AFRL-IF-RS-TR-2006-180
Final Technical Report
May 2006

SEARCH FOR NEW QUANTUM ALGORITHMS

University of Maryland Baltimore County (UMBC)

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. L485

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2006-180 has been reviewed and is approved for publication

APPROVED: /s/

STEVEN L. DRAGER
Project Engineer

FOR THE DIRECTOR: /s/

JAMES A. COLLINS, Deputy Chief
Advanced Computing Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
May 2006

2. REPORT TYPE
Final

3. DATES COVERED *(From – To)*
Jun 01 – Dec 05

4. TITLE AND SUBTITLE

SEARCH FOR NEW QUANTUM ALGORITHMS

5a. CONTRACT NUMBER
F30602-01-2-0522

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62712E

6. AUTHOR(S)

Samuel J. Lomonaco and Louis H. Kauffman

5d. PROJECT NUMBER
L485

5e. TASK NUMBER
SN

5f. WORK UNIT NUMBER
QA

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Maryland Baltimore County
1000 Hilltop Circle
Baltimore MD 21250

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

DARPA/IPTO
3701 N. Fairfax Dr
Arlington VA 22203

AFRL/IFTC
525 Brooks Rd
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2006-180

12. DISTRIBUTION AVAILABILITY STATEMENT

Approved for public release; distribution unlimited. PA# 06-356

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The first objective of this effort, searching for new quantum algorithms, created six new quantum hidden subgroup algorithms. The second objective, improving the theoretical understanding of existing quantum algorithms, produced three new systematic procedures. Also, application of combinatorial group theory led to substantial progress in the understanding and analysis of non-abelian quantum hidden subgroup algorithms. Additionally, methods and techniques of quantum topology have been used to obtain new results in quantum computing including discovery of a relationship between quantum entanglement and topological linking. The last objective, analyzing issues associated with algorithm implementation proposed distributed quantum computing (DQC) as a fast track to scalable quantum computing with technology available within the next five years. A universal set of DQC primitives has been created and used to transform the quantum Fourier transform and the Shor algorithm into DQC. The additional computational overhead needed for DQC algorithms is insignificant and DQC is found to simplify the decoherence problem.

15. SUBJECT TERMS

quantum algorithms, quantum hidden subgroup algorithms, hidden subgroups, distributed quantum computation, distributed algorithms, Shor's algorithm, Grover's algorithm, non-abelian quantum hidden subgroup algorithms, decoherence, quantum computer architectures, topological quantum computation, quantum gates, anyonic computation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Steven L. Drager
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPONE NUMBER (*Include area code*)
U	U	U	UL	56	

Table of Contents

Table of Figures

I. Executive summary of accomplishments relative to SOW

In the Statement of Work (SOW) mutually agreed upon with DARPA, we were tasked with the following:

- **Task 1.** Search for new quantum algorithms
- **Task 2.** Improve theoretical understanding of existing quantum algorithms
- **Task 3.** Analyze issues associated with implementation

In regard to **Task 1**, we have accomplished the following:

- We have created six new quantum hidden subgroup (QHS) algorithms:
    - A continuous Shor algorithm
    - The wandering Shor algorithm
    - The lifted Shor algorithm
    - A quantum circle algorithm
    - A QHS algorithm dual to Shor's algorithm
    - A QHS algorithm for functional integrals

In regard to **Task 2**, we have accomplished the following:

- In creating the above algorithms, we have found three new systematic procedures for creating and finding new quantum algorithms, namely:
    - Lifting
    - Pushing
    - Duality

- By applying the methods of combinatorial group theory, we have made substantial progress in the theoretical understanding and analysis of non-abelian quantum hidden subgroup (QHS) algorithms. In particular,

    - We have shown that non-abelian QHS algorithms can be lifted to quantum algorithms on free groups in polytime, and then pushed back to their original domain also in polytime. Hence, there is no loss of generality resulting from focusing only on non-abelian QHS algorithms on free groups, where the technical problems are substantially simpler. This is a significant result.

    - We also have shown that Schreier 2-sided transversals provide a best way to lift non-abelian QHS algorithms to free groups. This is a generalization of what we call the Shor transversal, which was used to create the wandering Shor algorithm.

- We have demonstrated that Grover's and Shor's quantum algorithms are more closely related than previously thought.

    - We have shown that Grover's algorithm is a QHS algorithm in the sense that it solves a non-abelian quantum hidden subgroup problem.
    - We have also shown that this QHS problem can not be solved by the standard non-abelian QHS algorithm.

- We have used the methods and techniques of quantum topology to obtain new results in quantum computing. In particular,
    - We have found all 4x4 unitary solutions of the Yang-Baxter equation, and determined which of these are universal quantum computing gates.
    - We have found relationships between quantum entanglement and topological linking.
    - We have determined new universal quantum gates that are solutions with spectral parameters of the Yang-Baxter equation.
    - We have shown how quantum teleportation can be understood in terms of the categorical formalism of quantum topology. This approach shows promise of being a useful tool for distributed quantum computing.
    - We have found a new approach to creating unitary representations of the braid group based on the bracket model of the Jones polynomial. This approach includes Kitaev's Fibonnaci model, and shows promise of yielding new insight into anyonic topological quantum computation.
    - We have classified all representations of the three-strand braid group.
    - We are currently writing a paper that shows that the quantum algorithm created by Aharonov, Freedman, Kitaev, Jones, and Landau for finding the values of the Jones polynomial can not successfully be extended via polynomial interpolation to a quantum algorithm that actually computes the full Jones polynomial.
    - We have developed a new combinatorial criterion for determining whether or not a pure state is entangled.

In regard to **Task 3**, we have accomplished the following:

- We have proposed distributed quantum computing (DQC) as a roadmap to scalable quantum computing, i.e., as a strategy for effectively using *existing* or near future (within five years) quantum computing devices (hardware) to perform BIG tasks, normally thought only possible on non-existent large quantum computers. By DQC, we mean quantum computing on a network of small quantum computers interconnected by quantum Einstein Poololsky Rosen (EPR) and classical channels. Any existing quantum computer device which, for example, can transform photon (flying) qubits into system qubits and back (such as ion traps, neutral atom devices, linear optics, etc.) could be used to form such a quantum network. We have obtained the following results in DQC:

- We have created a universal set of quantum distributive computing primitives for transforming existing quantum algorithms into distributed quantum algorithms, namely:
    - Cat-Creator
    - Disentangler
    - Reset
    - Swap-Reset

- Based on the above DQC primitives, we have created a systematic methodology for transforming non-distributed quantum algorithms into distributed quantum algorithms. This procedure can be automated.

- We have used the above mentioned systematic methodology to create a distributed version of the quantum Fourier transform and of Shor's quantum factoring algorithm.

- We have shown that distributed quantum algorithms are not significantly less efficient than their non-distributed counterparts. In particular, we have shown that the increase in computational complexity (resulting from distributed computing overhead) is insignificant in comparison with the greater efficiency and speed of quantum algorithms.

- We have also pointed out that DQC provides a mechanism for better dealing with the problem of decoherence, i.e., it provides a "divide and conquer" strategy for dealing with decoherence. Once EPR channels have been established, one need only focus on the decoherence problem for each spatially separated quantum device in the network. The key idea is that NOT all environmentally entangling transformations are equally likely. In particular, for spatially separated physical quantum computing devices, the most likely environmentally entangling transformations are those that are isolated to the local quantum device and its immediate environment. (This observation is debated by some in the quantum computing community.)

Also in regard to all **Tasks 1, 2, 3**, we have accomplished the following:

- As a research tool, we have developed a Maple software package for simulating quantum systems. This software, as well as an instruction manual, can be found at the website:
  http://www.csee.umbc.edu/~lomonaco/DARPA/01-06finalrpt

- Other research related to all tasks, but either not yet completed or not yet undertaken, is the following:

    - The use of QHS algorithms to extract geometric patterns from photos.

    - The design of DQC algorithms for systolic arrays of quantum smidgets, where by a quantum smidget, we mean a small but powerful quantum computing device, such as a quantum linear feedback shift register.

II. Accomplishments

II.A. Contributions to quantum algorithms

II.A.1. Definition of quantum hidden subgroup algorithms

What is a hidden subgroup problem? What is a hidden subgroup algorithm?

Definition. *A map $\varphi : G \to S$ from a group G into a set S is said to have **hidden subgroup structure** if there exists a subgroup K_φ of G, called a **hidden subgroup**, and an injection $\iota_\varphi : G / K_\varphi \to S$, called a **hidden injection**, such that the diagram*

$$\begin{tikzcd} G \arrow[rr, "\varphi"] \arrow[dr, "\nu"'] & & S \\ & G / K_\varphi \arrow[ur, "\iota_\varphi"'] & \end{tikzcd}$$

*is commutative[1], where G / K_φ denotes the collection of right cosets of K_φ in G, and where $\nu : G \to G / K_\varphi$ is the natural surjection of G onto G / K_φ. We refer to the group G as the **ambient group** and to the set S as the **target set**. If K_φ is a normal subgroup of G, then $H_\varphi = G / K_\varphi$ is a group, called the **hidden quotient group**, and $\nu : G \to G / K_\varphi$ is an epimorphism, called the **hidden epimorphism**. We will call the above diagram the **hidden subgroup structure** of the map $\varphi : G \to S$. (See [9, 59].)*

Remark. The underlying intuition motivating this formal definition is as follows: Given a natural surjection (or epimorphism) $\nu : G \to G / K_\varphi$, a "villain with malice of forethought" hides the algebraic structure of ν by intentionally renaming all the elements of G / K_φ, and "tossing in for good measure" some extra elements to form a set S and a map $\varphi : G \to S$.

The hidden subgroup problem can be stated as follows:

[1] By saying that this diagram is commutative, we m5ean $\varphi = \iota_\varphi \circ \nu$.

Hidden Subgroup Problem (HSP). *Let $\varphi : G \to S$ be a map with hidden subgroup structure. The problem of determining a hidden subgroup K_φ of G is* ***called a hidden subgroup problem (HSP)****. An algorithm solving this problem is called a* ***hidden subgroup algorithm****.*

The corresponding quantum form of this HSP is stated as follows:

Hidden Subgroup Problem (Quantum Version). *Let $\varphi : G \to S$ be a map with hidden subgroup structure. Construct a quantum implementation of the map φ as follows: Let H_G and H_S be Hilbert spaces defined respectively by the orthonormal bases $\{|g\rangle : g \in G\}$ and $\{|s\rangle : s \in S\}$ and let $s_0 = \varphi(1)$, where 1 denotes the identity[2] of the ambient group A. Finally, let U_φ be a unitary transformation such that*

$$\begin{array}{ccc} H_G \otimes H_S & \xrightarrow{U_\varphi} & H_G \otimes H_S \\ |g\rangle|s_0\rangle & \mapsto & |g\rangle|\varphi(g)\rangle \end{array}$$

Determine the hidden subgroup K_φ with bounded probability of error by making as few queries as possible of the blackbox U_φ. A quantum algorithm solving this problem is called a ***quantum hidden subgroup (QHS) algorithm****.*

II.A.2. The generic QHS algorithm

Let $\varphi : G \to S$ be a map from a group G to a set S with hidden subgroup structure. We assume that all representations of G are equivalent to unitary representations[3]. Let $\widehat{G}$ denote a **complete set of distinct irreducible unitary representations** of G. Using additive notation for G, we let 1 denote the **identity** of G, and let s_0 denote its image in S . Finally, let $\hat{1}$ denote the **trivial representation** of G.

Remark. If G is abelian, then $\widehat{G}$ becomes the **dual group** of characters.

The generic QHS algorithm is given below:

[2] We are using multiplicative notation for the group G.

[3] This is true for all finite groups as well as for a large class of infinite groups.

Quantum Subroutine $\mathit{QRand}(\varphi)$

Step 0. Initialization

$$|\psi_0\rangle = |\hat{1}\rangle |s_0\rangle \in \mathcal{H}_{\widehat{G}} \otimes \mathcal{H}_S$$

Step 1. Application of the inverse Fourier transform F_G^{-1} of G to the left register

$$|\psi_1\rangle = \frac{1}{\sqrt{|G|}} \sum_{g\in G} |g\rangle |s_0\rangle \in \mathcal{H}_G \otimes \mathcal{H}_S \ ,$$

where $|G|$ denotes the cardinality of the group G.

Step 2. Application of the unitary transformation U_φ

$$|\psi_2\rangle = \frac{1}{\sqrt{|G|}} \sum_{g\in G} |g\rangle |\varphi(g)\rangle \in \mathcal{H}_G \otimes \mathcal{H}_S$$

Step 3. Application of the Fourier transform F_G of G to the left register

$$|\psi_3\rangle = \frac{1}{|G|} \sum_{\gamma\in\widehat{G}} |\gamma| \, \mathit{Trace}\left(\sum_{g\in G} \gamma^\dagger(g) |\gamma\rangle \right) |\varphi(g)\rangle = \frac{1}{|G|} \sum_{\gamma\in\widehat{G}} |\gamma| \, \mathit{Trace}\left(|\gamma\rangle |\Phi(\gamma^\dagger)\rangle \right) \in \mathcal{H}_{\widehat{G}} \otimes \mathcal{H}_S \ ,$$

where $|\gamma|$ denotes the **degree** of the representation γ, where $\gamma^\dagger$ denotes the **contragradient representation** (i.e., $\gamma^\dagger(g) = \gamma(g^{-1})^T = \overline{\gamma(g)}^T$), where $\mathit{Trace}(\gamma^\dagger |\gamma\rangle) = \sum_{i=1}^{|\gamma|} \sum_{j=1}^{|\gamma|} \overline{\gamma_{ji}(g)} |\gamma_{ij}\rangle$, and where $|\Phi(\gamma_{ij}^\dagger)\rangle = \sum_{g\in G} \overline{\gamma_{ji}(g)} |\varphi(g)\rangle$.

Step 4. Measurement of the left quantum register with respect to the orthonormal basis

$$\left\{ |\gamma_{ij}\rangle : \gamma \in \widehat{G}, \ 1 \le i,j \le |\gamma| \right\}.$$

Thus, with probability

$$\mathbf{Prob}_\varphi(\gamma_{ij}) = \frac{|\gamma|^2 \langle \Phi(\gamma_{ij}^\dagger) | \Phi(\gamma_{ij}^\dagger) \rangle}{|G|^2}$$

the resulting measured value is the entry γ_{ij}, and the quantum system "collapses" to the state

$$|\psi_4\rangle = \frac{|\gamma_{ij}\rangle |\Phi(\gamma_{ij}^\dagger)\rangle}{\sqrt{\langle \Phi(\gamma_{ij}^\dagger) | \Phi(\gamma_{ij}^\dagger) \rangle}} \in \mathcal{H}_{\widehat{G}} \otimes \mathcal{H}_S$$

Step 5. Output γ_{ij}, and stop.

II.A.3. Shor's factoring algorithm

Briefly, Shor's factoring algorithm is described as follows:

Let N be the integer to be factored. Choose an integer Q such that $N^2 \leq Q = 2^k < 2N^2$, and a random integer ℓ which with high probability is relatively prime to N. Let $\varphi : \mathbb{Z} \to \mathbb{Z}_N$ be the map, defined by the $n \mapsto \ell^n \bmod N$, from the additive group of integers $\mathbb{Z}$ into the integers mod N (under multiplication), denoted by $\mathbb{Z}_N$. Shor's algorithm proceeds to find the hidden subgroup $P\mathbb{Z}$ of $\mathbb{Z}$ of all multiples of P, where P is the unknown period of the map φ.

This is accomplished by constructing the epimorphism $\nu : \mathbb{Z} \to \mathbb{Z}_Q$ from the additive group of integers $\mathbb{Z}$ to the additive group of integers $\mathbb{Z}_Q$ modulo Q defined by $\nu : n \mapsto n \bmod Q$, a transversal[4] $\iota : \mathbb{Z}_Q \to \mathbb{Z}$ for the epimorphism ν, and a map $\widetilde{\varphi} = \varphi \circ \iota : \mathbb{Z}_Q \to \mathbb{Z}_N$ which is a "good approximation" to the map φ.

$$\begin{tikzcd}
P\mathbb{Z} \arrow[r, phantom, "\subset"] & \mathbb{Z} \arrow[r, "\varphi"] \arrow[d, shift right, "\nu"'] & \mathbb{Z}_N \\
& \mathbb{Z}_Q \arrow[u, shift right, "\iota"'] \arrow[ur, "\widetilde{\varphi} = \varphi \circ \iota"'] &
\end{tikzcd}$$

Then Shor's algorithm calls the quantum subroutine **QRand**$(\widetilde{\varphi})$ to produce a random rational d/Q which is with high probability close to a rational of the form y/P. By "d/Q close to y/P," we mean that d/Q is a convergent of the continued fraction expansion of the rational y/P. With high probability, the integers y and P are relatively prime, and thus, the unknown period P is found.

Remark. The mathematically alert reader will recognize the rational d/Q as the character $\chi_{d/Q} : \mathbb{Z}_Q \to \mathbb{R} \bmod 1$ of $\mathbb{Z}_Q$ defined by $n \bmod Q \mapsto e^{2\pi i n d/Q}$, and the rational y/P as the character $\chi_{y/P} : \mathbb{Z}_P \to \mathbb{R} \bmod 1$ defined by $n \bmod P \mapsto e^{2\pi i n y/P}$.

[4] By a **transversal** of ν, we mean an injection ι such that $\iota \circ \nu = id_Q$ is the identity map on $\mathbb{Z}_Q$. In other words, a transversal maps each element of $\mathbb{Z}_Q$ into an element of the coset of $\mathbb{Z}$ representing that coset.

II.A.4. Wandering Shor algorithms, a.k.a., vintage shor algorithms

Wandering Shor algorithms are essentially QHS algorithms on free abelian finite rank n group A which, with each iteration, first select a random cyclic direct summand $\mathbb{Z}$ of the group A, and then apply one iteration of the standard Shor algorithm to produce a random character of the "approximating" finite group $\widetilde{A} = \mathbb{Z}_Q$, called a **group probe**[5]. Three different wandering Shor algorithms are created in [9]. The first two wandering Shor algorithms given in [9] are quantum algorithms which find the order P of a maximal cyclic subgroup of the hidden quotient group H_φ. The third computes the entire hidden quotient group H_φ.

The first step in creating a wandering Shor algorithm is to find the right generalization of the transversal $\iota : \mathbb{Z}_Q \to \mathbb{Z}$ found in Shor's factoring algorithm, i.e., to construct the correct transversal from $\mathbb{Z}_Q$ to a free abelian group A of rank n. For this reason, we have created the following definition:

Definition. *Let A be the free abelian group of rank n, let $\mathbb{Z}_Q$ be the cyclic group of order Q, and let $\tilde{a}$ denote a chosen generator of $\mathbb{Z}_Q$. An injection $\iota : \mathbb{Z}_Q \to A$ is said to be a* ***Shor injection*** *provided that:*

- $\iota\left(n\tilde{a}\right) = n\iota\left(\tilde{a}\right)$ *for all* $0 \le n < Q$
- *For each (free abelian) basis* $a_1', a_2', \ldots, a_n'$ *of* A*, the coefficients* $\lambda_1', \lambda_2', \ldots, \lambda_n'$ *of* $\iota\left(\tilde{a}\right) = \sum_j \lambda_j' a_j'$ *satisfy* $\gcd\left(\lambda_1', \lambda_2', \ldots, \lambda_n'\right) = 1$.

Remark. Later, when we construct the right generalization of Shor transversals to free groups of finite rank n, we will see that a Shor transversal is nothing more than a 2-sided Schreier transversal. Hence, we will see then that the second condition of the above definition simply says that ι maps the generator $\tilde{a}$ of $\mathbb{Z}_Q$ onto a generator of a free direct summand $\mathbb{Z}$ of A. (For more details, please refer to section II.A.9 of this report.)

Remark. In [9], we show how to use the extended Euclidean algorithm to construct an epimorphism $\nu : A \to \mathbb{Z}_Q$ having $\iota : \mathbb{Z}_Q \to A$ as its transversal. Consequently, we call ι a **Shor transversal** for the epimorphism ν.

[5] By a group probe $\widetilde{A}$, we mean an epimorphic image of the ambient group A.

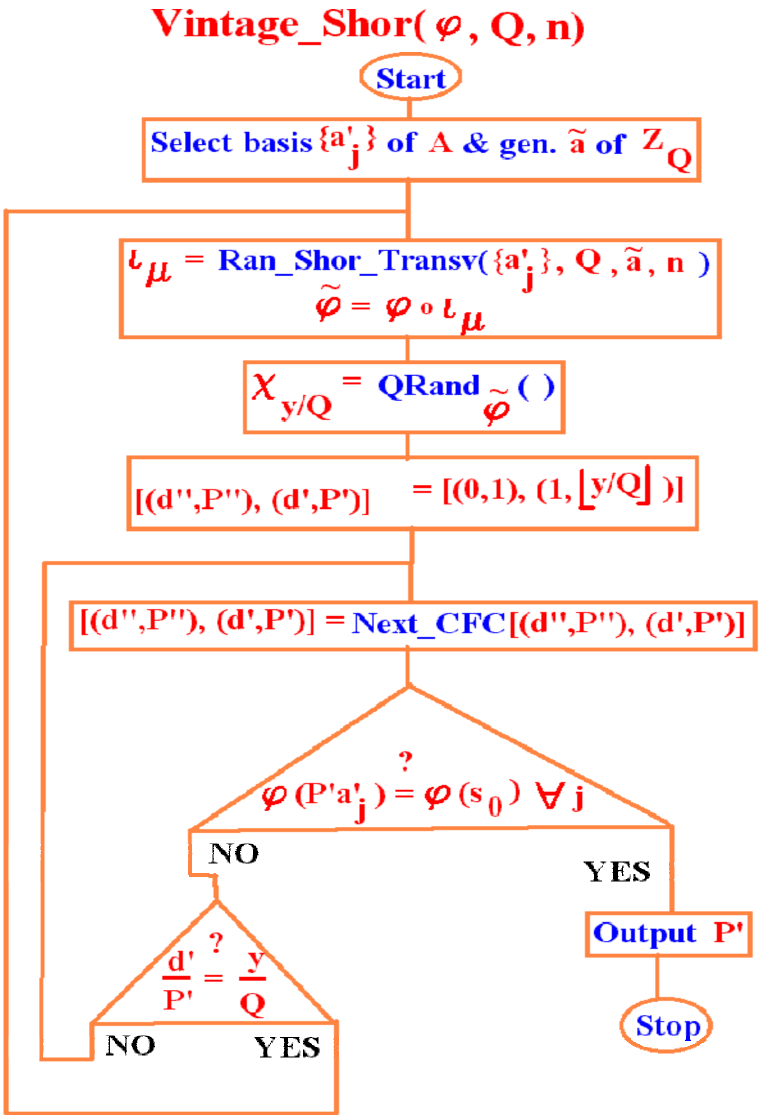

Vintage_Shor(φ, Q, n)
Start
Select basis {a'_j} of A & gen. ã of Z_Q
ι_μ = Ran_Shor_Transv({a'_j}, Q , ã , n)
φ̃ = φ ∘ ι_μ
χ_{y/Q} = QRand_φ̃ ( )
[(d'',P''), (d',P')] = [(0,1), (1, ⌊y/Q⌋)]
[(d'',P''), (d',P')] = Next_CFC[(d'',P''), (d',P')]
φ (P'a'_j) =? φ (s_0) ∀ j
NO
YES
Output P'
Stop
d'/P' =? y/Q
NO
YES

Figure 1. Flowchart for the first wandering Shor algorithm (a.k.a., a vintage Shor algorithm). This algorithm finds the order P of a maximal cyclic subgroup of the hidden quotient group H_φ.

Flow charts for the three wandering Shor algorithms created in [9] are given in figures 1 through 3. In [9], these were also called **vintage Shor algorithms**.

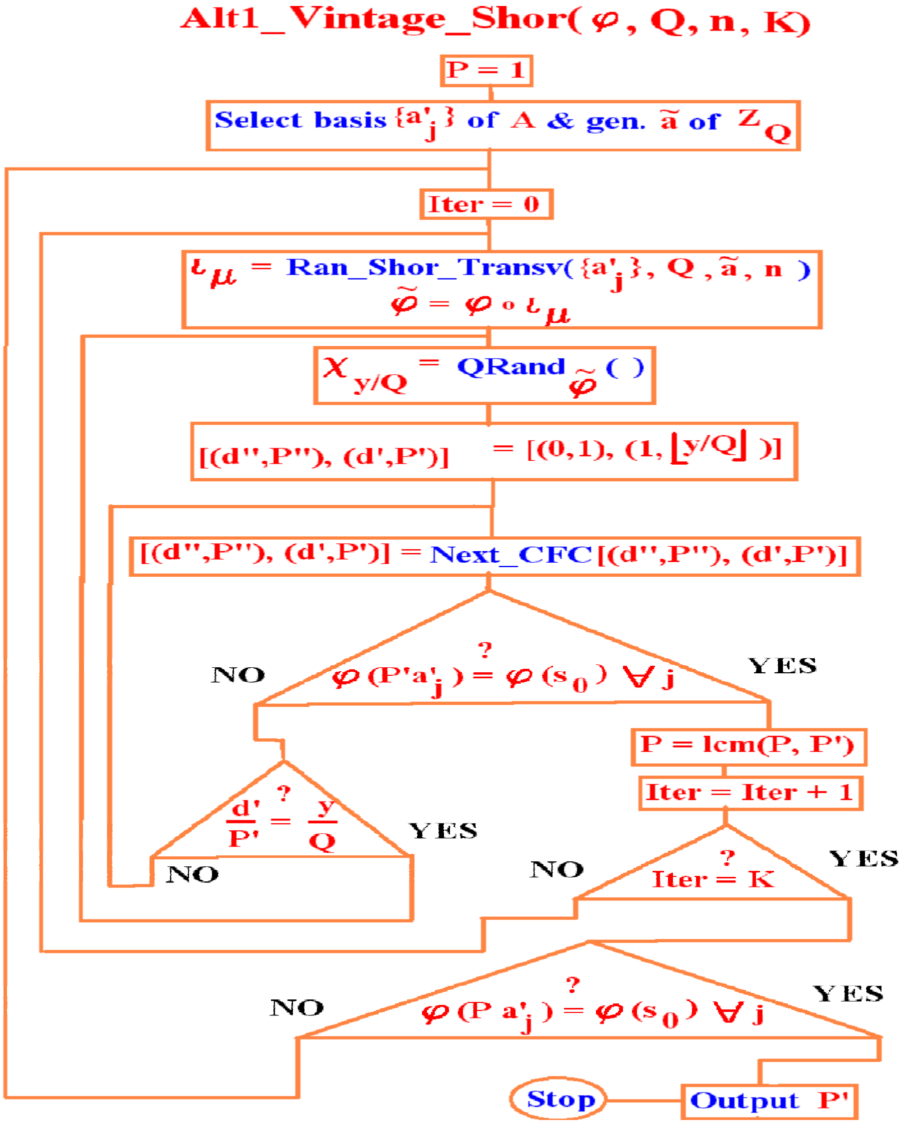

Alt1_Vintage_Shor(φ, Q, n, K)
P = 1
Select basis {a'j} of A & gen. ã of ZQ
Iter = 0
ιμ = Ran_Shor_Transv({a'j}, Q , ã, n)
φ̃ = φ ∘ ιμ
χy/Q = QRand φ̃ ( )
[(d'',P''), (d',P')] = [(0,1), (1, ⌊y/Q⌋)]
[(d'',P''), (d',P')] = Next_CFC [(d'',P''), (d',P')]
φ (P'a'j) =? φ (s0) ∀ j
NO
YES
P = lcm(P, P')
Iter = Iter + 1
d'/P' =? y/Q
YES
NO
Iter =? K
NO
YES
φ (P a'j) =? φ (s0) ∀ j
NO
YES
Stop
Output P'

Figure 2. Flowchart for the second wandering Shor algorithm (a.k.a., a vintage Shor algorithm). This algorithm finds the order P of a maximal cyclic subgroup of the hidden quotient group H_φ.

The algorithmic complexities of the above wandering Shor algorithms is given in [9]. For example, the first wandering Shor algorithm is of time complexity

$$O\left(n^2 (\lg N)^3 (\lg\lg N)^{n+1}\right),$$

where n is the rank of the free abelian group A. This can be readily deduced from the following abbreviated flowchart given in figure 4.

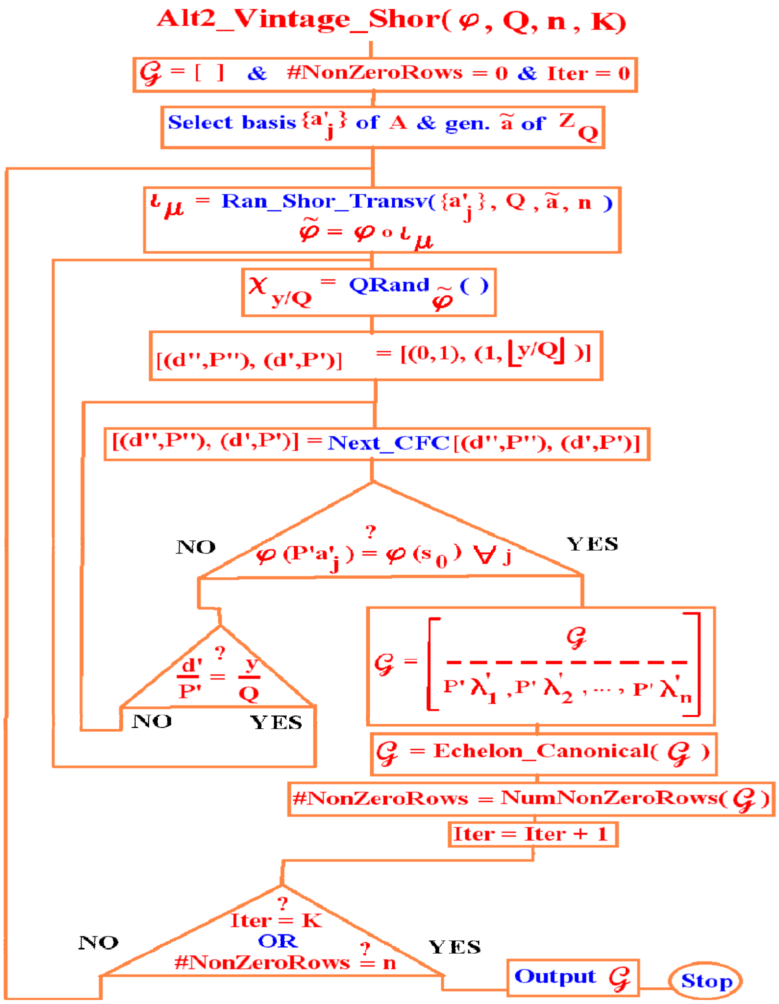

Alt2_Vintage_Shor(φ, Q, n , K)
𝒢 = [ ] & #NonZeroRows = 0 & Iter = 0
Select basis {a'_j} of A & gen. ã of Z_Q
ι_μ = Ran_Shor_Transv({a'_j}, Q , ã, n)
φ̃ = φ ∘ ι_μ
χ_y/Q = QRand_φ̃ ( )
[(d'',P''), (d',P')] = [(0,1), (1, ⌊y/Q⌋)]
[(d'',P''), (d',P')] = Next_CFC[(d'',P''), (d',P')]
NO
φ (P'a'_j) =? φ (s_0) ∀ j
YES
d'/P' =? y/Q
NO
YES
𝒢 = [𝒢 / P' λ'_1 , P' λ'_2 , ... , P' λ'_n]
𝒢 = Echelon_Canonical(𝒢)
#NonZeroRows = NumNonZeroRows(𝒢)
Iter = Iter + 1
NO
Iter =? K OR #NonZeroRows =? n
YES
Output 𝒢
Stop

Figure 3. Flowchart for the third wandering Shor algorithm, a.k.a., a vintage Shor algorithm. This algorithm finds the entire hidden quotient group H_φ.

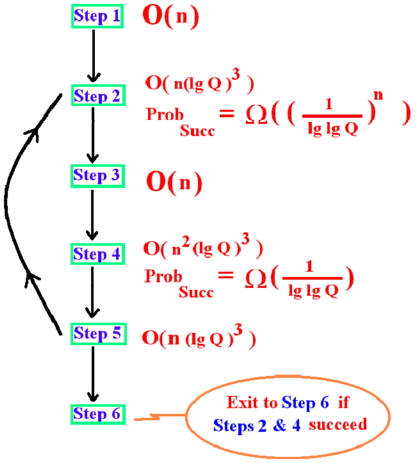

Step 1
O(n)
Step 2
O(n(lg Q)^3)
Prob_Succ = Ω((1 / lg lg Q)^n)
Step 3
O(n)
Step 4
O(n^2 (lg Q)^3)
Prob_Succ = Ω(1 / lg lg Q)
Step 5
O(n (lg Q)^3)
Step 6
Exit to Step 6 if Steps 2 & 4 succeed

Figure 4. Abbreviated flowchart for the first wandering Shor algorithm.

II.A.5. A continuous (variable) Shor algorithm

A continuous variable Shor algorithm was created in [13, 24]. By a **continuous variable Shor algorithm**, we mean a quantum hidden subgroup algorithm that finds the hidden period P of an admissible function $\varphi : \mathbb{R} \to \mathbb{R}$ from the reals $\mathbb{R}$ to itself.

Remark. By an admissible function, we mean a function belonging to any sufficiently well behaved class of functions. For example, the class of functions which are Lebesgue integrable on every closed interval of $\mathbb{R}$. There are many other classes of functions that work equally as well.

Actually, the papers [13, 24] give in succession three such continuous Shor algorithms, each successively more general than the previous.

For the first algorithm, we assume that the hidden period P is an integer. The algorithm is then constructed by using rigged Hilbert spaces, linear combinations of Dirac delta functions, and a subtle extension of the Fourier transform in the generic QHS subroutine **QRand(φ)**, which has been described previously in section II.A.2 of this report. In Step 5 of this algorithm, the observable

$$A = \int_{-\infty}^{\infty} dy \frac{\lfloor Qy \rfloor}{Q} |y\rangle\langle y|$$

is measured, where Q is an integer chosen so that $Q \geq 2P^2$. It then follows that the output of this algorithm is a rational m/Q which is a convergent of the continued fraction expansion of a rational of the form n/P.

The above quantum algorithm is then extended to a quantum algorithm (the second algorithm) that finds the hidden period P of functions $\varphi : \mathbb{R} \to \mathbb{R}$, when P is a rational.

Finally, the second algorithm is extended to the third algorithm which finds the hidden period P of functions $\varphi : \mathbb{R} \to \mathbb{R}$, when P is an *arbitrary real number*. We point out for the third and last algorithm to work, we must impose a very restrictive condition on the map $\varphi : \mathbb{R} \to \mathbb{R}$, i.e., the condition that the map φ is continuous.

II.A.6. Three techniques for creating new QHS algorithms, i.e., lifting, pushing, and duality

We have already seen one ingredient of Shor's factoring algorithm that we were able to generalize to create new QHS algorithms. Specifically, we have generalized the transversal $\iota : \mathbb{Z}_Q \to \mathbb{Z}$ found in Shor's original algorithm to a transversal $\iota : \mathbb{Z}_Q \to A$ (called a **Shor transversal**) from the finite cyclic group $\mathbb{Z}_Q$ to the free abelian group A of finite rank. This enabled us to create new quantum algorithms, called wandering Shor algorithms. (See section II.A.4.) Later, in section II.A.9, we will see how this ingredient can be further generalized to a **2-sided Schreier transversal**, thereby enabling us to create new QHS algorithms on free groups.

In this section, we seek to generalize other ingredients found within Shor's original algorithm with the intention of using these generalizations to create even more new quantum algorithms. For that reason, we give the following definitions:

Definition. *A map $\widetilde{\varphi}:\widetilde{G}\rightarrow S$ from a group $\widetilde{G}$ to a set S is said to be the **push** of a map $\varphi:G\rightarrow S$ from a group G to S, written*

$$\widetilde{\varphi}=Push(\varphi)$$

if there exists an epimorphism $\nu:G\rightarrow\widetilde{G}$ of G onto $\widetilde{G}$ and a transversal $\iota:\widetilde{G}\rightarrow G$ of ν such that $\widetilde{\varphi}=\varphi\circ\iota$.

Thus, in terms of the above definition, we can now see that Shor's algorithm, as described in section II.A.3., is an algorithm created by **pushing** the hidden subgroup problem (HSP) $\varphi:\mathbb{Z}\rightarrow S$ to $Push(\varphi)=\widetilde{\varphi}:\mathbb{Z}_Q\rightarrow\mathbb{Z}_N$, and then calling $QRand(\widetilde{\varphi})$ to produce a character $\chi_{y/Q}$ of the group $\mathbb{Z}_Q$. The character $\chi_{y/Q}$ is with high probability close to a primitive character $\chi_{d/P}$ of the hidden subgroup $\mathbb{Z}_P$. Hence, the character $\chi_{d/P}$ can be found with the standard continued fraction algorithm.

We next note that the above generic definition of pushing suggests a second procedure which can in turn also be used for creating even more QHS algorithms, namely:

Definition. *A map $\varphi:G\rightarrow S$ from a group G to a set S is said to be the **lift** of a map $\widetilde{\varphi}:\widetilde{G}\rightarrow S$ from a group $\widetilde{G}$ to S, written*

$$\varphi=Lift(\widetilde{\varphi})$$

if there exists a morphism $\mu:G\rightarrow\widetilde{G}$ of G into $\widetilde{G}$ such that $\varphi=\widetilde{\varphi}\circ\mu$.

There is yet a third generic procedure that we have used to create new QHS algorithms -- duality.

Let G be an abelian group, and let $\widehat{G}$ denote its dual group of characters. Then if we are creating QHS algorithms for hidden subgroup problems (HSPs) of the form $\varphi:G\rightarrow S$, we might as well use the same procedures to create QHS algorithms for HSPs of the form $\varphi:\widehat{G}\rightarrow S$. We refer to this method of creating new QHS algorithms as **duality**.

In the sections to follow, we will show how lifting, pushing, and duality are used to create new algorithms. In particular, in section II.A.7, we use lifting and duality to create three new QHS algorithms, namely, the lifted Shor algorithm, the circle algorithm, and the dual Shor algorithm. In section II.A.9, we will use pushing and lifting to create QHS algorithms on free groups.

II.A.7. The lifted Shor, the quantum circle, and the dual Shor algorithms

The methods of lifting and duality given in the previous section are used in [13, 19, 24] to create the following QHS algorithms:

- The lifted Shor algorithm -- Created from the Shor algorithm by lifting
- The circle algorithm -- Created from the lifted Shor algorithm by duality
- The dual Shor algorithm -- Created from the circle algorithm by lifting

A roadmap for creating these algorithms is given in figure 5.

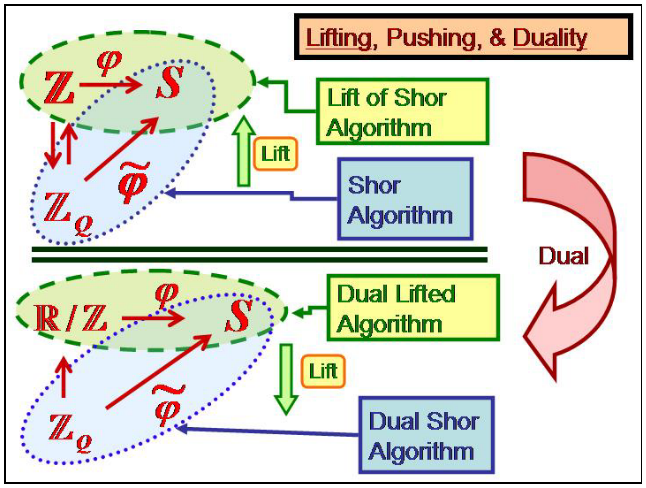


Figure 5. Roadmap for creating new QHS algorithms.

As its name suggests, the **lifted Shor algorithm** is created by **lifting** the hidden subgroup problem (HSP) $\widetilde{\varphi}:\mathbb{Z}_Q \to \mathbb{Z}_N$ to a HSP $\varphi:\mathbb{Z} \to \boldsymbol{S}$, thereby producing a QHS algorithm which is essentially a "distillation" of Shor's original algorithm. Next, **duality** is used to create the **quantum circle algorithm** by devising a QHS algorithm for the HSP $\varphi:\mathbb{R}/\mathbb{Z} \to \boldsymbol{S}$ on the dual group $\mathbb{R}/\mathbb{Z}$ of the additive group of integers $\mathbb{Z}$. (By $\mathbb{R}/\mathbb{Z}$, we mean the **additive group of reals mod 1**, which is isomorphic to the multiplicative group $\left\{e^{i\theta} : 0 \le \theta < 2\pi\right\}$, i.e., the unit **circle** in the complex plane.) Finally, the **dual Shor algorithm** is created by **lifting** the HSP $\varphi:\mathbb{R}/\mathbb{Z} \to \boldsymbol{S}$ to the HSP $\widetilde{\varphi}:\mathbb{Z}_Q \to \boldsymbol{S}$.

It is not clear whether or not the lifted Shor algorithm and the quantum circle algorithm[6] are physically implementable quantum algorithms. Even so, they play an important role as two essential stepping stones to the very implementable dual Shor algorithm.

For detailed descriptions of each of these quantum algorithms, i.e., the **lifted Shor, the quantum circle, and the dual Shor** algorithms, the reader is referred to [13, 19, 24].

We give below brief descriptions of the quantum circle and the dual Shor algorithms.

For the **quantum circle algorithm**, we make use of the following spaces (each of which is used in quantum optics)

- The rigged Hilbert space $\boldsymbol{H}_{\mathbb{R}/\mathbb{Z}}$ with orthonormal basis $\{|x\rangle : x \in \mathbb{R}/\mathbb{Z}\}$. By "orthonormal" we mean that $\langle x \mid y\rangle = \delta(x-y)$, where "$\delta$" denotes the Dirac delta function. The elements of $\boldsymbol{H}_{\mathbb{R}/\mathbb{Z}}$ are **formal integrals** of the form $\oint dx\, f(x)|x\rangle$. (The physicist Dirac in his classic book on quantum mechanics refers to these integrals as infinite sums.)
- The complex vector space $\boldsymbol{H}_{\mathbb{Z}}$ of formal sums
$$\left\{ \sum_{n=-\infty}^{\infty} a_n |n\rangle : a_n \in \mathbb{C} \quad \forall n \in \mathbb{Z} \right\}$$
with orthonormal basis $\{|n\rangle : n \in \mathbb{Z}\}$. By "orthonormal" we mean that $\langle n \mid m\rangle = \delta_{nm}$, where δ_{nm} denotes the Kronecker delta.

We will now design an algorithm which solves the following hidden subgroup problem:

Hidden Subgroup Problem for the Circle. *Let $\varphi : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$ be an admissible function from the circle group $\mathbb{R}/\mathbb{Z}$ to the complex numbers $\mathbb{C}$ with hidden rational period $\alpha \in \mathbb{Q}/\mathbb{Z}$, where $\alpha \in \mathbb{Q}/\mathbb{Z}$ denotes the rational circle, i.e., the rationals $\mathbf{mod}\ 1$.*

Remark. By an admissible function, we mean a function belonging to any sufficiently well behaved class of functions. For example, the class of functions which are Lebesgue integrable on $\mathbb{R}/\mathbb{Z}$. There are many other classes of functions that work equally as well.

Proposition. *Let $\alpha = a_1/a_2$ (with $\gcd(a_1, a_2) = 1$) be a period of a function $\varphi : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$. Then $1/a_2$ is also a period of φ. Hence, the minimal rational period of φ is always a reciprocal integer $\mathbf{mod}\ 1$.*

[6] There is a possibility that the quantum circle algorithm may have a physical implementation in terms of quantum optics.

The following quantum algorithm finds the reciprocal integer period of the function φ.

Circle-Algorithm(φ)

Step 0. Initialization

$$\left|\psi_0\right\rangle = \left|0\right\rangle \left|0\right\rangle \in H_{\mathbb{Z}} \otimes H_{\mathbb{C}}$$

Step 1. Application of the inverse Fourier transform $F^{-1} \otimes 1$

$$\left|\psi_1\right\rangle = \oint dx\, e^{2\pi i \bullet 0} \left|x\right\rangle \left|0\right\rangle = \oint dx \left|x\right\rangle \left|0\right\rangle \in H_{\mathbb{R}/\mathbb{Z}} \otimes H_{\mathbb{C}}$$

Step 2. Application of the unitary transformation $U_\varphi : \left|x\right\rangle \left|u\right\rangle \mapsto \left|x\right\rangle \left|u + \varphi(x)\right\rangle$

$$\left|\psi_2\right\rangle = \oint \left|x\right\rangle \left|\varphi(x)\right\rangle \in H_{\mathbb{R}/\mathbb{Z}} \oplus H_{\mathbb{C}}$$

Step 3. Application of the Fourier transform $F \otimes 1$

$$\left|\psi_2\right\rangle = \sum_{n \in \mathbb{Z}} \oint dx\, e^{-2\pi i n x} \left|n\right\rangle \left|\varphi(x)\right\rangle = \sum_{n \in \mathbb{Z}} \left|n\right\rangle \oint dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle \in H_{\mathbb{Z}} \oplus H_{\mathbb{C}}$$

Remark. *Letting* $x_m = x - \dfrac{m}{a}$, *we have*

$$\begin{aligned}\oint dx\, e^{2\pi i n x} \left|\varphi(x)\right\rangle &= \sum_{m=0}^{a-1} \int_{m/a}^{(m+1)/a} dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle \\ &= \sum_{m=0}^{a-1} \int_{0}^{1/a} dx_m\, e^{-2\pi i n \left(x_m + \frac{m}{a}\right)} \left|\varphi\left(x_m + \frac{m}{a}\right)\right\rangle \\ &= \left(\sum_{m=0}^{a-1} e^{-2\pi i n m/a}\right) \int_0^{1/a} dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle\end{aligned}$$

where $1/a$ *is the unknown reciprocal period. But*

$$\sum\nolimits_{m=0}^{a-1} e^{-2\pi i n m/a} = a\delta_{n=0 \bmod a} = \begin{cases} a & \textit{if } n = 0 \bmod a \\ 0 & \textit{otherwise} \end{cases}$$

Hence,

$$\left|\psi_3\right\rangle = \sum_{n \in \mathbb{Z}} \left|n\right\rangle \oint dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle = \left(\sum_{n \in \mathbb{Z}} \left|n\right\rangle \delta_{n=0 \bmod a}\right) \int_0^{1/a} dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle$$

$$= \left(\sum_{\ell \in \mathbb{Z}} \left|\ell a\right\rangle\right) \left(\int_0^{1/a} dx\, e^{-2\pi i n x} \left|\varphi(x)\right\rangle\right) = \sum_{\ell \in \mathbb{Z}} \left|\ell a\right\rangle \left|\Omega(\ell a)\right\rangle$$

Step 4. Measurement of

$$\left|\psi_3\right\rangle = \sum_{\ell\in\mathbb{Z}} \left|\ell a\right\rangle \left|\Omega(\ell a)\right\rangle \in H_{\mathbb{Z}} \otimes H_{\mathbb{C}}$$

with respect to the observable

$$\sum_{n\in\mathbb{Z}} n\left|n\right\rangle\left\langle n\right|$$

to produce a random eigenvalue ℓa .

Remark. The above quantum circle algorithm can be extended to a quantum algorithm which finds the hidden period P of a function $\varphi : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$, when P is an *arbitrary real number* $\mathbf{mod\ 1}$. But in creating this extended quantum algorithm, we must impose a very restrictive condition on the map $\varphi : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$, i.e., the condition that the map φ is continuous.

We now give a brief description of the **dual Shor algorithm**.

The dual Shor algorithm is a QHS algorithm created by making a discrete approximation of the quantum circle algorithm. More specifically, it is created by lifting the QHS circle algorithm for $\varphi : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$ to the finite cyclic group $\mathbb{Z}_Q$, as illustrated in the commutative diagram given below:

$$\begin{tikzcd}
\mathbb{Z}_Q \arrow[d, "\mu"'] \arrow[dr, "\widetilde{\varphi} = Push(\varphi) = \varphi \circ \mu"] & \\
\mathbb{R}/\mathbb{Z} \arrow[r, "\varphi"] & S
\end{tikzcd}$$

Intuitively, we "approximate" the circle group $\mathbb{R}/\mathbb{Z}$ with the finite cyclic group $\mathbb{Z}_Q$ by identifying $\mathbb{Z}_Q$ with the following additive group

$$\left\{\frac{0}{Q}, \frac{1}{Q}, \ldots, \frac{Q-1}{Q}\right\} \mathbf{mod\ 1},$$

and by identifying the hidden subgroup $\mathbb{Z}_P$ with the additive group

$$\left\{\frac{0}{P}, \frac{1}{P}, \ldots, \frac{P-1}{P}\right\} \mathbf{mod\ 1},$$

where $P = a_2$.

This is a physically implementable quantum algorithm. It is actually faster than Shor's algorithm. For the last step of Shor's algorithm uses the standard continued fraction algorithm to determine the unknown period. On the other hand, the last step of the dual Shor algorithm uses the much faster Euclidean algorithm to compute the greatest common divisor of the integers $\ell_1 a, \ell_2 a, \ell_3 a, \ldots$, thereby determining the desired reciprocal integer period $1/a$. For more details, please refer to [13, 19, 14].

II.A.8. A QHS algorithm for Feynman integrals

We now discuss a QHS algorithm based on Feynman path integrals. This quantum algorithm was developed at the Mathematical Sciences Research Institute (MSRI) in Berkeley, California when the PI was challenged by being invited to give a talk on Feynmann path integrals and quantum computing at an MSRI conference on Feynman path integrals.

Until recently, the PI thought that the quantum algorithm to be described below was a highly speculative quantum algorithm, because the existence of Feynman path integrals is very difficult (if not impossible) to determine in a mathematically rigorous fashion. But much to the PI's surprise, Jeremy Becnel in his doctoral dissertation [36] has succeeded in creating a firm mathematical foundation for this algorithm.

We should mention, however, that the physical implementability of this algorithm is still yet to be determined.

Definition. *Let* $\boldsymbol{Paths}$ *be the real vector space of all continuous paths* $\boldsymbol{x}:[0,1]\to\mathbb{R}^n$ *which are* $\boldsymbol{L}^2$ *with respect to the inner product*

$$x\bullet y=\int_0^1 ds\; x(s)y(s)$$

with scalar multiplication and vector sum defined as

- $(\lambda x)(s)=\lambda x(s)$
- $(x+y)(s)=x(s)+y(s)$

We wish to create a QHS algorithm for the following hidden subgroup problem:

Hidden Subgroup Problem for $\boldsymbol{Paths}$**.** *Let* $\varphi:\boldsymbol{Paths}\to\mathbb{C}$ *be a functional with a hidden subspace* V *of* $\boldsymbol{Paths}$ *such that*

$$\varphi(x+v)=\varphi(x)\quad\forall v\in V$$

Our objective is to create a QHS algorithm which solves the above problem, i.e., which finds the hidden subspace V.

Definition. *Let* $\boldsymbol{H}_{Paths}$ *be the rigged Hilbert space with orthonormal basis* $\{|x\rangle : x\in \boldsymbol{Paths}\}$, *and with bracket product* $\langle x\,|\,y\rangle=\delta(x-y)$.

We will use the following observation to create the QHS algorithm:

Observation. $\mathit{Paths} = \bigcup_{v \in V} \left(v + V^{\perp} \right)$, *where* $V^{\perp}$ *denotes the orthogonal complement of the hidden vector subspace* V.

The QHS algorithm for Feynman path integral is now given below:

Feynman (φ)

Step 0. Initialize

$$\left| \psi_0 \right\rangle = \left| \mathbf{0} \right\rangle \left| 0 \right\rangle \in \mathrm{H}_{\mathit{Paths}} \otimes \mathrm{H}_{\mathbb{C}}$$

Step 1. Apply $F^{-1} \otimes 1$

$$\left| \psi_1 \right\rangle = \int_{\mathit{Paths}} \boldsymbol{D}x \quad e^{2\pi i x \bullet 0} \left| x \right\rangle \left| 0 \right\rangle = \int_{\mathit{Paths}} \boldsymbol{D}x \left| x \right\rangle \left| 0 \right\rangle$$

Step 2. Apply $U_{\varphi} : \left| x \right\rangle \left| u \right\rangle \mapsto \left| x \right\rangle \left| u + \varphi(x) \right\rangle$

$$\left| \psi_2 \right\rangle = \int_{\mathit{Paths}} \boldsymbol{D}x \left| x \right\rangle \left| \varphi(x) \right\rangle$$

Step 3. Apply $F \otimes 1$

$$\begin{aligned} \left| \psi_3 \right\rangle &= \int_{\mathit{Paths}} \boldsymbol{D}y \int_{\mathit{Paths}} \boldsymbol{D}x \, e^{-2\pi i x \bullet y} \left| y \right\rangle \left| \varphi(x) \right\rangle \\ &= \int_{\mathit{Paths}} \boldsymbol{D}y \left| y \right\rangle \int_{\mathit{Paths}} \boldsymbol{D}x \, e^{-2\pi i x \bullet y} \left| \varphi(x) \right\rangle \end{aligned}$$

But

$$\begin{aligned} \int_{\mathit{Paths}} \boldsymbol{D}x \, e^{-2\pi i x \bullet y} \left| \varphi(x) \right\rangle &= \int_{V} \boldsymbol{D}v \int_{v+V^{\perp}} \boldsymbol{D}x \, e^{-2\pi i x \bullet y} \left| \varphi(x) \right\rangle \\ &= \int_{V} \boldsymbol{D}v \int_{V^{\perp}} \boldsymbol{D}x \, e^{-2\pi i (v+x) \bullet y} \left| \varphi(v+x) \right\rangle \\ &= \int_{V} \boldsymbol{D}v \, e^{-2\pi i v \bullet y} \int_{V^{\perp}} \boldsymbol{D}x \, e^{-2\pi i x \bullet y} \left| \varphi(x) \right\rangle \end{aligned}$$

However,

$$\int_{V} \boldsymbol{D}v \, e^{-2\pi i v \bullet y} = \int_{V^{\perp}} \boldsymbol{D}u \, \delta(y - u)$$

So,

$$\begin{aligned}|\psi_3\rangle &= \int_{Paths_n} \mathcal{D}y\,|y\rangle \int_{V} \mathcal{D}v\, e^{-2\pi i v\cdot y} \int_{V^\perp} \mathcal{D}x\, e^{-2\pi i x\cdot y}\,|\varphi(x)\rangle \\ &= \int_{Paths_n} \mathcal{D}y\,|y\rangle \int_{V^\perp} \mathcal{D}u\,\delta(y-u) \int_{V^\perp} \mathcal{D}x\, e^{-2\pi i x\cdot y}\,|\varphi(x)\rangle \\ &= \int_{V^\perp} \mathcal{D}u\,|u\rangle \int_{V^\perp} \mathcal{D}x\, e^{-2\pi i x\cdot u}\,|\varphi(x)\rangle \\ &= \int_{V^\perp} \mathcal{D}u\,|u\rangle|\Omega(u)\rangle\end{aligned}$$

Step 4. Measure

$$|\psi_3\rangle = \int_{V^\perp} \mathcal{D}u\,|u\rangle|\Omega(u)\rangle$$

with respect to the observable

$$A = \int_{Paths} \mathcal{D}w\; w|w\rangle\langle w|$$

to produce a random element of $V^\perp$

The above algorithm suggests an intriguing question. Can the above QHS Feynman integral algorithm be modified in such a way as to create a quantum algorithm for the Jones polynomial? In other words, can it be modified by replacing ***Paths*** with the space of gauge connections, and making suitable modifications?

This question is motivated by the fact that the integral over gauge transformation

$$\widehat{\psi}(K) = \int \mathcal{D}A\,\psi(A)W_K(A)$$

looks very much like a Fourier transform, where

$$W_K(A) = tr\left(P\exp\left(\oint_K A\right)\right)$$

denotes the Wilson loop over the knot K.

II.A.9. Non-abelian QHS algorithms -- A simplification

Let $\varphi : G \to S$ be a map with hidden subgroup structure from a finitely generated (f.g.) non-abelian group G to a set S. We assume that the hidden subgroup K is a normal subgroup of G.

Under the above assumptions, we now outline why one need only study non-abelian QHS algorithms on free groups. More specifically, we briefly describe the following results:

- Every hidden subgroup problem $\varphi : G \longrightarrow S$ on a non-abelian f.g. group G can be lifted to a hidden subgroup problem $\widetilde{\varphi} : F \longrightarrow S$ on a f.g. free group F, and in turn
- Every polytime QHS algorithm for $\widetilde{\varphi} : F \longrightarrow S$ can be transformed via the Reidemeister-Schreier theorem into a polytime QHS algorithm for the original hidden subgroup problem $\varphi : G \longrightarrow S$.

But what do we mean by a free group?

(Universal) Definition. *A f.g. group F is said to be **free** if there exists a set of generators $X = \left\{ x_1, x_2, \ldots, x_n \right\}$ such that, for every group G and for every map $f : X \longrightarrow G$ of the set X into the group G, the map f extends to a morphism $\widetilde{f} : F \longrightarrow G$. We call the set X a **free basis** of the group F, and frequently denote the group F by $F\left(x_1, x_2, \ldots, x_n \right)$. It follows from this definition that the morphism $\widetilde{f}$ is unique.*

The intuitive idea encapsulated by this definition is that a free group is an unconstrained group (very much analogous to a physical system without boundary conditions.) In other words, a group is free provided it has a set of generators such that the only relations among those generators are those required for F to be a group. For example,

- $x_i x_i^{-1} = 1$ is an allowed relation
- $x_i x_j = x_j x_i$ is not an allowed relation for $i \neq j$
- $x_i^3 = 1$ is not an allowed relation

Definition. *Let $F\left(x_1, x_2, \ldots, x_n \right)$ be a free group with free basis $x_1, x_2, \ldots, x_n$. Then a **word** is a finite string of the symbols $x_1, x_1^{-1}, x_2, x_2^{-1}, \ldots, x_n, x_n^{-1}$. A **reduced word** is a word in which there is no substring of the form $x_j x_j^{-1}$ or $x_j^{-1} x_j$. Two words are said to be **equivalent** if one can be transformed into the other by applying a finite number of substring insertions or deletions of the form $x_j x_j^{-1}$ or $x_j^{-1} x_j$. We denote an **arbitrary word** w by $w = a_1 a_2 \cdots a_\ell$, where each $a_j = x_{k_j}^{\pm 1}$.*

For example, $x_2 x_1^{-1} x_1 x_1^{-1} x_5^{-1} x_5^{-1} x_5^{-1} x_5$ is a word which is equivalent to the reduced word $x_2 x_1^{-1} x_5^{-1} x_5^{-1}$.

It easily follows that a free group $F(x_1,x_2,\ldots,x_n)$ is nothing more than the set of reduced words together with the obvious definition of product, i.e., concatenation with full reduction.

Definition. *Let G be a group. A **group presentation***

$$(x_1,x_2,\ldots,x_n : r_1,r_2,\ldots,r_m)$$

*for G is a set of free generators $x_1,x_2,\ldots,x_n$ of a free group F and a set of words $r_1,r_2,\ldots,r_n$ in $F(x_1,x_2,\ldots,x_n)$, called **relators**, such that the group G is isomorphic to the quotient group $F(x_1,x_2,\ldots,x_n)/Cons(r_1,r_2,\ldots,r_n)$, where $Cons(r_1,r_2,\ldots,r_n)$, called the **consequence** of $r_1,r_2,\ldots,r_n$, is the smallest normal subgroup of $F(x_1,x_2,\ldots,x_n)$ containing the relators $r_1,r_2,\ldots,r_n$.*

The intuition captured by the above definition is that $x_1,x_2,\ldots,x_n$ are the generators of G, and $r_1=1, r_2=1, \ldots, r_n=1$ is a complete set of relations among these generators, i.e., every relation among the generators of G is a **consequence** (derivable from) the relations $r_1=1, r_2=1, \ldots, r_n=1$. For example,

- $(x_1,x_2,\ldots,x_n :)$ and $(x_1,x_2,\ldots,x_n : x_1x_1^{-1}, x_2{}^5x_2{}^{-5}, x_3x_4x_4{}^{-1}x_3{}^{-1})$ are both presentations of the free group $F(x_1,x_2,\ldots,x_n)$
- $(x : x^Q)$ and $(x : x^a, x^b)$ are both presentations of the cyclic group $\mathbb{Z}_Q$ of order Q, where a and b are integers such that $\gcd(a,b)=Q$.
- $(x_1,x_2 : x_1{}^3, x_2{}^2, (x_1x_2)^2)$ is a presentation of the symmetric group S_3 on three symbols.

The procedure we are about to describe makes use of the **Reidemeister-Schreier (R-M) theorem**, a well known theorem in combinatorial group theory. Given a presentation $(x_1,x_2,\ldots,x_n : r_1,r_2,\ldots,r_m)$ of a group G and a subgroup K of G, the R-M theorem gives an algorithm for computing a presentation of the subgroup K. This R-M theorem algorithm is constructed using a Schreier transversal.

But what is a Schreier transversal?

Definition. *A set W of reduced words in a free group $F=F(x_1,x_2,\ldots,x_n)$ is said to be a **2-sided Schreier system** provided*

- $w=a_1a_2\cdots a_{\ell-1}a_\ell \in W \Rightarrow w_{Left}=a_1a_2\cdots a_{\ell-1}\in W$, *and*
- $w=a_1a_2\cdots a_{\ell-1}a_\ell \in W \Rightarrow w_{Right}=a_2\cdots a_{\ell-1}a_\ell \in W$

Given an epimorphism $\nu : F \to G$ *of the free group* F *onto a group* G*, a* ***2-sided Schreier transversal*** $\tau : G \to F$ *for* ν *is a transversal for* ν *for which there exists a 2-sided Schreier system such that* $\tau(G) = W$.

Remark. *Most surprisingly, the concept of a 2-sided Schreier transversal provides a very natural way of extending Shor's original factoring algorithm to a quantum algorithm on free groups. For the transversal found in Shor's original quantum algorithm and in the wandering Shor's algorithm are nothing more than abelian 2-sided Schreier transversals! (See sections II.A.3 and II.A.4.)*

Finally, we are now prepared to describe the central idea of this section.

Given the hidden subgroup problem (HSP) $\varphi : G \to S$ with hidden subgroup K, we construct a QHS algorithm for φ using the following procedure:

STEP 1. Construct a presentation $(x_1, \ldots, x_n : r_1, \ldots, r_m)$ for the group G, and let $\nu : F(x_1, x_2, \ldots, x_m) \to G$ denote the epimorphism corresponding to this presentation.

STEP 2. Construct the lifted HSP $\widetilde{\varphi} = Lift(\varphi) = \varphi \circ \nu : F \to S$. (See section II.A.6.)

STEP 3. Use $QRand(\widetilde{\varphi})$ to find the hidden subgroup $\widetilde{K} = \nu^{-1}(K)$ of the HSP $\widetilde{\varphi}$. (See section II.A.7.)

STEP 4. Construct a 2-sided Schreier transversal $\tau : G \to F$ for ν .

STEP 5. With the above transversal τ , use the R-M Theorem to construct a presentation of the hidden subgroup $\widetilde{K}$ of the original HSP φ.

More details about this approach to creating new non-abelian QHS algorithms can be found in the forthcoming paper [32].

II.A.10. Is Grover's algorithm a QHS algorithm?

Is Grover's algorithm a quantum hidden subgroup algorithm? Does Grover's algorithm have some symmetries that we can exploit?

The problem solved by Grover's algorithm [7, 51, 52, 53] is that of finding an unknown integer label j_0 in an unstructured database with items labeled by the integers:

$$0,1,2,\ldots,j_0,\ldots,N-1=2^n-1,$$

given the oracle

$$f(j)=\begin{cases}1 & if\ j=j_0\\ 0 & otherwise\end{cases}$$

Let H be the Hilbert space with orthonormal basis $|0\rangle,|1\rangle,|2\rangle,\ldots,|N-1\rangle$. Grover's oracle is essentially given by the unitary transformation

$$\begin{array}{rcl}I_{|j_0\rangle}:H & \to & H\\ |j\rangle & \mapsto & (-1)^{f(j)}|j\rangle\end{array}$$

where $I_{|j_0\rangle}=I-2|j_0\rangle\langle j_0|$ is inversion in the hyperplane orthogonal to $|j\rangle$. Let W denote the Hadamard transformation on the Hilbert space H. Then Grover's algorithm is as follows:

<u>STEP</u> 0. (Initialization)

$$\begin{array}{lcl}|\psi\rangle & \leftarrow & W|0\rangle=\dfrac{1}{\sqrt{N}}\displaystyle\sum_{j=0}^{N-1}|j\rangle\\ k & \leftarrow & 0\end{array}$$

<u>STEP</u> 1. Loop until $k\approx\pi\sqrt{N}/4$

$$\begin{array}{lcl}|\psi\rangle & \leftarrow & Q|\psi\rangle=-WI_{|0\rangle}WI_{|j_0\rangle}|\psi\rangle\\ k & \leftarrow & k+1\end{array}$$

<u>STEP</u> 2. Measure $|\psi\rangle$ with respect to the standard basis

$$|0\rangle,|1\rangle,|2\rangle,\ldots,|N-1\rangle$$

to obtain the unknown state $|j_0\rangle$ with

$$\text{Prob}\geq 1-\frac{1}{N}$$

But where is the hidden symmetry in Grover's algorithm?

Let S_N be the symmetric group on the symbols $0,1,2,\ldots,N-1$. Then Grover's algorithm is invariant under the **hidden subgroup** $Stab_{j_0}=\{g\in S_N : g(j_0)=j_0\}\subset S_N$, called the **stabilizer subgroup** for j_0, i.e., Grover's algorithm is invariant under the group action

$$\begin{array}{ccc} Stab_{j_0} \times H & \rightarrow & H \\ \left(g, \sum_{j=0}^{N-1} a_j \left| j \right\rangle \right) & \mapsto & \sum_{j=0}^{N-1} a_j \left| g(j) \right\rangle \end{array}$$

Moreover, if we know the hidden subgroup $Stab_{j_0}$, then we know j_0, and vice versa. In other words, the problem of finding the unknown label j_0 is informationally the same as the problem of finding the hidden subgroup $Stab_{j_0}$.

Let $(ij) \in S_N$ denote the permutation that interchanges integers i and j, and leaves all other integers fixed. Thus, (ij) is a transposition if $i \neq j$, and the identity permutation 1 if $i = j$.

Proposition. The set $\left\{ (0\ j_0), (1\ j_0), (2\ j_0), \ldots, ((N-1)\ j_0) \right\}$ *is a complete set of distinct coset representatives for the hidden subgroup* $Stab_{j_0}$ *of* S_N*, i.e., the coset space* $S_N / Stab_{j_0}$ *is given by the following complete set of distinct cosets:*

$$S_N / Stab_{j_0} = \left\{ (0\ j_0) Stab_{j_0}, (1\ j_0) Stab_{j_0}, (2\ j_0) Stab_{j_0}, \ldots, ((N-1)\ j_0) Stab_{j_0} \right\}$$

We can now see that Grover's algorithm is a hidden subgroup algorithm in the sense that it is a quantum algorithm which solves the following hidden subgroup problem:

Grover's Hidden Subgroup Problem. *Let* $\varphi : S_N \rightarrow S$ *be a map from the symmetric group* S_N *to a set* $S = \{0, 1, 2, \ldots, N-1\}$ *with hidden subgroup structure given by the commutative diagram*

$$\begin{tikzcd} S_N \arrow[rr, "\varphi"] \arrow[dr, "\nu_{j_0}"'] & & S \\ & S_N / Stab_{j_0} \arrow[ur, "\iota"'] & \end{tikzcd},$$

where $\nu_{j_0} : S_N \rightarrow S_N / Stab_{j_0}$ *is the natural surjection of* S_N *on to the coset space* $S_N / Stab_{j_0}$*, and where*

$$\begin{array}{cccc} \iota : & S_N / Stab_{j_0} & \rightarrow & S \\ & (j\ j_0) Stab_{j_0} & \mapsto & j \end{array}$$

is ***the unknown relabeling*** *(bijection) of the coset space* $S_N / Stab_{j_0}$ *onto the set* S*. Find the hidden subgroup* $Stab_{j_0}$ *with bounded probability of error.*

Now let us compare Shor's algorithm with Grover's.

From section **II.A.3**, we know that Shor's algorithm [5, 9, 71, 72] solves the hidden subgroup problem $\varphi : \mathbb{Z} \to \mathbb{Z}_N$ with hidden subgroup structure

$$\begin{tikzcd}
\mathbb{Z} \arrow[rr, "\varphi"] \arrow[dr, "\nu"'] & & \mathbb{Z}_N \\
& \mathbb{Z}/P\mathbb{Z} \arrow[ur, "\iota"'] &
\end{tikzcd}$$

Moreover, as stated in section **II.A.3**, Shor creates his algorithm by pushing[7] the above hidden subgroup problem $\varphi : \mathbb{Z} \to \mathbb{Z}_N$ to the hidden subgroup problem $\widetilde{\varphi} : \mathbb{Z}_Q \to \mathbb{Z}_N$ (called Shor's oracle), where the hidden subgroup structure of $\widetilde{\varphi}$ is given by the commutative diagram

$$\begin{tikzcd}
\mathbb{Z} \arrow[rr, "\varphi"] \arrow[dr, shift right, "\alpha"'] & & \mathbb{Z}_N \\
& \mathbb{Z}_Q \arrow[ul, shift right, "\tau"'] \arrow[ur, "\widetilde{\varphi} = \varphi \circ \tau"'] &
\end{tikzcd} \quad ,$$

where α is the natural epimorphism of $\mathbb{Z}$ onto $\mathbb{Z}_Q$, and where τ is Shor's chosen transversal for the epimorphism α.

Surprisingly, Grover's algorithm, viewed as an algorithm that solves the Grover hidden subgroup problem, is very similar to Shor's algorithm.

Like Shor's algorithm, Grover's algorithm solves a hidden subgroup problem, i.e., the Grover hidden subgroup problem $\varphi : S_N \to S$ with hidden subgroup structure

$$\begin{tikzcd}
S_N \arrow[rr, "\varphi"] \arrow[dr, "\nu"'] & & S \\
& S_N / Stab_{j_0} \arrow[ur, "\iota"'] &
\end{tikzcd} \quad ,$$

where $S = \{0,1,2,\ldots,N-1\}$ denotes the set resulting from an unknown relabeling (bijection)

$$\left(j\ j_0\right) Stab_{j_0} \mapsto j$$

of the coset space

$$S_N / Stab_{j_0} = \left\{ \left(0\ j_0\right) Stab_{j_0}, \left(1\ j_0\right) Stab_{j_0}, \left(2\ j_0\right) Stab_{j_0}, \ldots, \left(\left(N-1\right) j_0\right) Stab_{j_0} \right\}.$$

Also, like Shor's algorithm, we can think of Grover's algorithm as one created by pushing the Grover hidden subgroup problem $\varphi : S_N \to S$ to the hidden subgroup

[7] See Section II.A.6 for a definition of pushing.

problem $\widetilde{\varphi} : S_N / Stab_{j_0} \to S$, where the pushing is defined by the following commutative diagram

$$\begin{tikzcd}
S_N \arrow[rr, "\varphi"] \arrow[dr, shift left, "\alpha"] & & S = S_N / Stab_{j_0} \\
& S_N / Stab_0 \arrow[ul, shift left, "\tau"] \arrow[ur, "\widetilde{\varphi} = \varphi \circ \tau"'] &
\end{tikzcd},$$

where $\alpha : S_N \to S_N / Stab_0$ denotes the natural surjection of S_N onto the coset space $S_N / Stab_0$, and where $\tau : S_N / Stab_0 \to S_N$ denotes the transversal of α given by

$$\begin{array}{ccc} S_N / Stab_0 & \to & S_N \\ (j\ 0)\,Stab_0 & \mapsto & (j\ 0) \end{array}.$$

Again also like Shor's algorithm, the map $\widetilde{\varphi}$ given by

$$\begin{array}{ccc} S_N / Stab_0 & \to & S_N / Stab_{j_0} = S \\ (j\ 0)\,Stab_0 & \mapsto & (j\ j_0)\,Stab_{j_0} = j \end{array}$$

is (if $j_0 \neq 0$) actually a disguised Grover's oracle. For the map $\widetilde{\varphi}$ can easily be shown to simply to

$$\widetilde{\varphi}\left((j\ 0)Stab_0\right) = \begin{cases} (j\ 0)Stab_{j_0} & if\ \ j = j_0 \\ Stab_{j_0} & otherwise \end{cases},$$

which is informationally the same as Grover's oracle

$$f(j) = \begin{cases} j & if\ \ j = j_0 \\ 1 & otherwise \end{cases}$$

Hence, we can conclude that Grover's algorithm is a quantum algorithm very much like Shor's algorithm, in that it is a quantum algorithm that solves the Grover hidden subgroup problem.

However, this appears to be where the similarity between Grover's and Shor's algorithms ends. For the standard non-abelian QHS algorithm for S_N cannot find the hidden subgroup $Stab_{j_0}$ for each of following two reasons:

- Since the subgroups $Stab_j$ are not normal subgroups of S_N, it follows from the work of Hallgren et al [55, 56] that the standard non-abelian hidden subgroup algorithm will find the largest normal subgroup of S_N lying in $Stab_{j_0}$. But unfortunately, the largest normal subgroup of S_N lying in $Stab_j$ is the trivial subgroup of S_N.
- The subgroups $Stab_0, Stab_1, \ldots, Stab_{N-1}$ are mutually conjugate subgroups of S_N.

Moreover, one can not hope to use this QHS approach to Grover's algorithm to find a faster quantum algorithm. For Zalka [77] has shown that Grover's algorithm is optimal.

The arguments given above suggest that Grover's and Shor's algorithms are more closely related than one might at first expect. Although the standard non-abelian QHS algorithm on $\boldsymbol{S}_N$ can not solve the Grover hidden subgroup problem, there does remain an intriguing question:

Question. *Is there some modification of (or extension of) the standard QHS algorithm on* $\boldsymbol{S}_N$ *that actually solves Grover's hidden subgroup problem?*

The results found within this section can be found in the forthcoming paper [31].

II.A.11 Distributed quantum computing (DQC)

How can we use current or near future technology (i.e., technology available now or within the next five years) to solve tasks that we normally think could only be solved by large quantum computers, which probably will not be available for at least twenty to thirty years into the future?

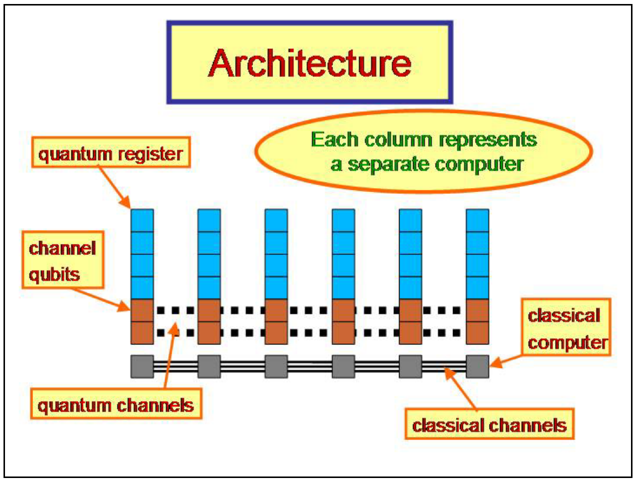


Figure 6. Computer architecture model for quantum distributed computing (QDC).

As an answer to this question, we have in [18, 25] proposed distributed quantum computing (DQC) as a fast track roadmap to scalable quantum computing, i.e., as a strategy for effectively using technology available now or within the next five years to perform BIG tasks, normally thought only possible on future technology twenty to thirty years down the road.

II.A.12. The key idea for DQC

By DQC, we mean quantum computing on a network of small quantum computers interconnected by quantum (EPR) and classical channels, as illustrated in the figure 6. Any existing or near future quantum computer device which, for example, can transform photon (flying) qubits into system qubits and back (such as ion traps, neutral atom devices, linear optics, etc.) could be used to form such a quantum network.

The key idea in [18, 25] is to use quantum entanglement to distribute control to the different computers within the above mentioned quantum network. In particular, our strategy is to use **generalized GHZ states** to create **cat-like states**, which in turn are to be used to distribute control.

By a **generalized GHZ state**, we mean a quantum state of the form

$$\frac{\left|\overbrace{00\cdots0}^{n\ \ 0's}\right\rangle+\left|\overbrace{11\cdots1}^{n\ \ 1's}\right\rangle}{\sqrt{2}},$$

and by a **cat-like state**, we mean a quantum state of the form

$$\frac{\alpha\left|00\cdots0\right\rangle+\beta\left|11\cdots1\right\rangle}{\sqrt{2}}.$$

One important point to emphasize is that once EPR channels have been established, each of the above entangled states is created by applying only local unitary operations to the individual computing nodes within the quantum network. Moreover, once a cat-like state has been created, it does not matter which of the cat-like state qubits is used for quantum control. They all do operationally the same thing. This is illustrated in figure 7.

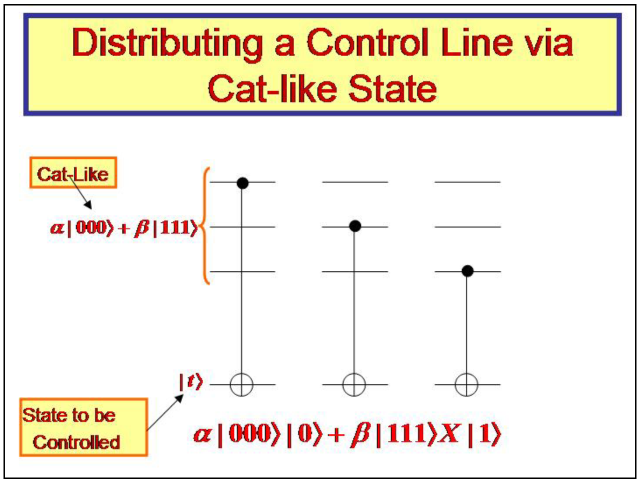

Distributing a Control Line via Cat-like State
Cat-Like
$\alpha|000\rangle+\beta|111\rangle$
$|t\rangle$
State to be Controlled
$\alpha|000\rangle|0\rangle+\beta|111\rangle X|1\rangle$

Figure 7. All qubits of a given cat-like state are operationally the same when it comes to quantum control..

An n qubit generalized GHZ state is created by the n qubit entangling gate shown in figure 8,

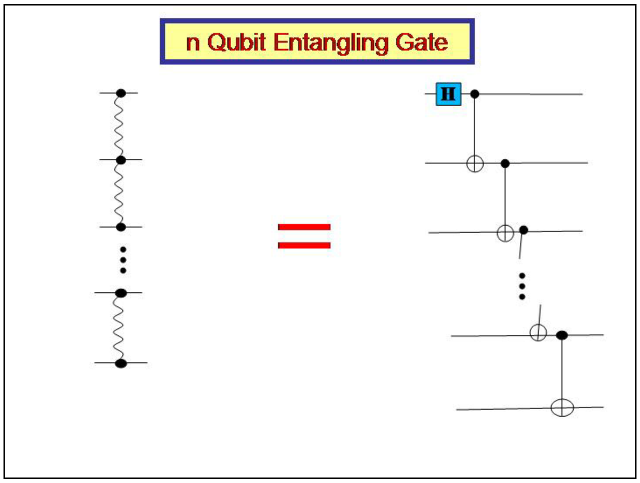


Figure 8. An n qubit entangling gate.

where H denotes the 1-qubit Hadamard transform, and where we have used the conventional notation for the controlled-NOT gate.

II.A.13. A universal set of DQC primitives

Next in [18, 25], we find a universal set of quantum distributive computing primitives for creating DQC algorithms, and show systematically how to convert existing quantum algorithms into distributed quantum algorithms. (This procedure can be automated.) In particular, these DQC primitives are:

- **Cat-Creator**
- **Disentangler**
- **Reset**
- **Swap-Reset**

Cat-Creator is shown in figure 9 below,

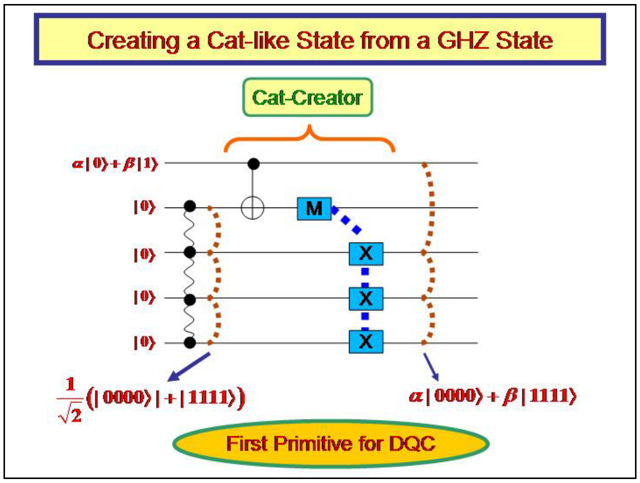


Figure 9. Cat-Creator transforms a generalized GHZ state into a cat-like state.

where "**M**" denotes a (standard basis) one qubit measurement gate, where "**X**" denotes the Pauli-X gate. The blue dashed line indicates that the classical bit produced by the measurement gate **M** is used to control the **X** gates.

The **Disentangler** primitive is shown in figure 10,

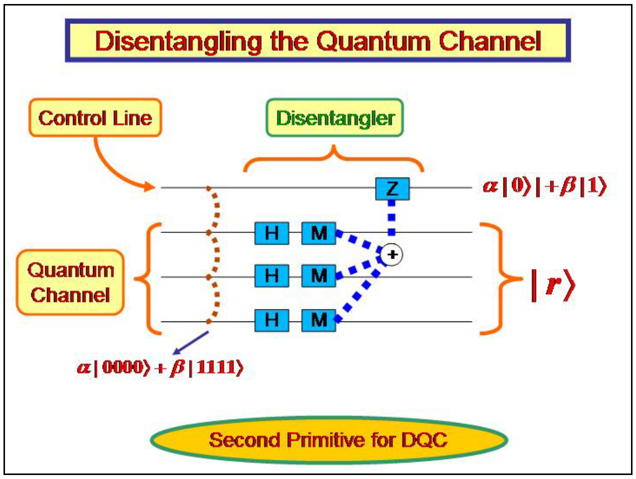
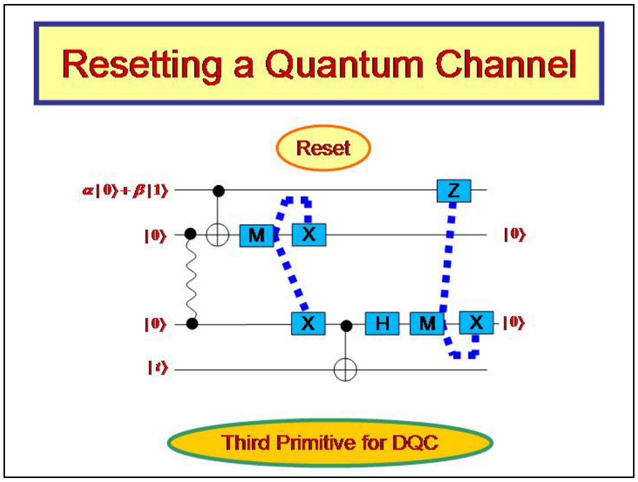


Figure 10. The Disentangler primitive systematically disassembles a cat-like state after it has been utilized.

where "**H**" again denotes a one qubit Hadamard gate, "**M**" a (standard basis) one qubit measuring gate, and "**Z**" a Pauli-Z gate. As indicated by the blue dashed line, the classical bits produced by the **M** gates are sent to a classical "logical OR" gate "⊕" whose output is in turn used to control the Pauli-Z gate.

The DQC primitives **Reset** and **Swap-Reset** are shown in figures 11 and 12, respectively.

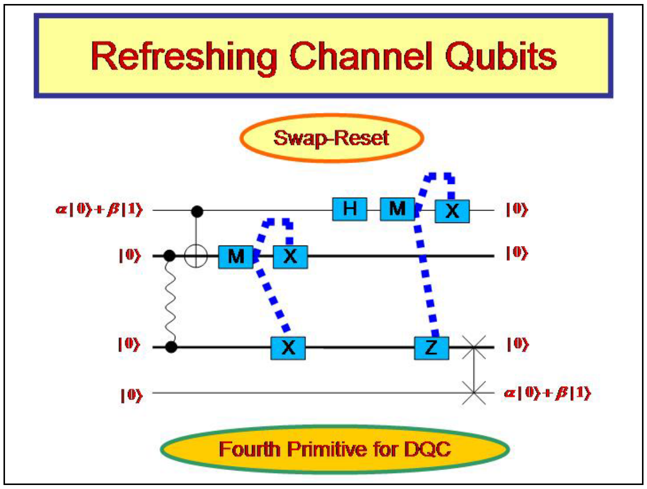


Resetting a Quantum Channel

Reset

Third Primitive for DQC

Figure 11. The QDC primitive Reset.



Refreshing Channel Qubits

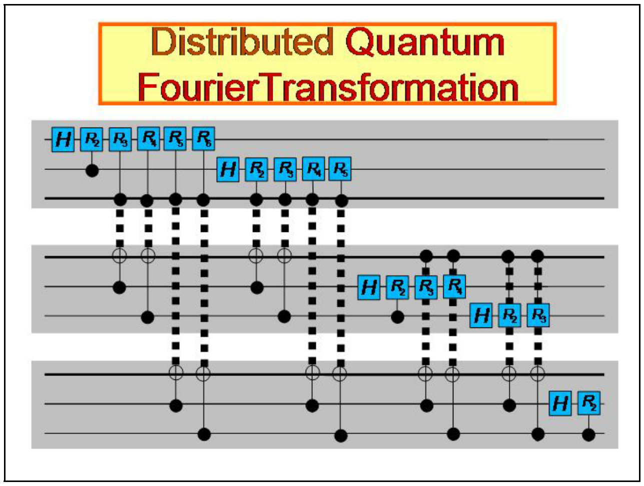
Swap-Reset

Fourth Primitive for DQC

Figure 12. The QDC primitive Swap-Reset.

II.A.14. A systematic procedure for creating distributed quantum algorithms

A systematic procedure is then given in [18,25] (which can be automated) for transforming quantum algorithms into distributed quantum algorithms. This systematic procedure is then used in [18,25] to create a distributed quantum Fourier transform, and also a distributed Shor factoring algorithm. As an example, a distributed quantum Fourier transform is shown in figure 13.

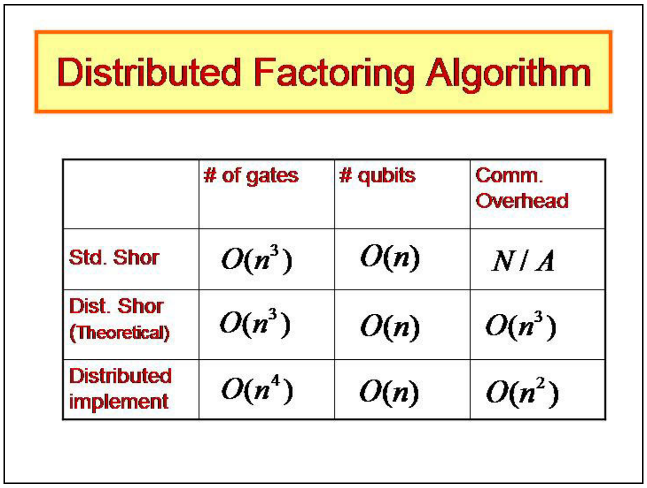


Distributed Quantum FourierTransformation

Figure 13. An example of a distributed quantum Fourier transform.

II.A.15. The computational overhead resulting from converting a quantum algorithm into a distributed one

We have also compared the complexity of QDC algorithms with their non-distributive quantum algorithm counterparts. A complexity comparison is given for Shor's algorithm in the table found in figure 14.



Distributed Factoring Algorithm

	# of gates	# qubits	Comm. Overhead
Std. Shor	$O(n^3)$	$O(n)$	N / A
Dist. Shor (Theoretical)	$O(n^3)$	$O(n)$	$O(n^3)$
Distributed implement	$O(n^4)$	$O(n)$	$O(n^2)$

Figure 14. Comparison of the non-distributed Shor algorithm with the distributed Shor algorithm.

The asymptotic bounds given in the above table are very conservative estimates. Even so, it is surprising and significant that the additional time and space complexity estimates for transforming a quantum algorithm into a distributed one are essentially washed out by the time and space complexity of the shor algorithm. In other words, for shor's algorithm, the additional computational overhead is insignificant! The same can be said for many other quantum algorithms. The worst time complexity overhead we have seen resulting from transforming examples of various quantum algorithms into distributed ones is a linear slowdown and a linear increase in the number of gates. All these results appear to support the case that DQC is a viable roadmap to scalable quantum computing.

II.A.16. DQC as a divide-and-conquer attack on the problem of decoherence

We have also pointed out that DQC provides a mechanism for better dealing with the problem of decoherence. It provides an opportunity for the application of a "divide and conquer" strategy for dealing with decoherence.

Once EPR channels have been established, one need only focus on the decoherence problem for each spatially separated quantum device in the network and its immediate environment. The key idea is that NOT all environmentally entangling transformations are equally likely. In particular, for spatially separated physical quantum computing devices, the most likely entangling transformations are those which are isolated to the local quantum device and its immediate environment. This is a substantial simplification.

Not everyone in the quantum computing community agrees with this assessment and observation. We believe this stems from a misunderstanding of what we have said above. What we are saying is simply that, at each node, one need only consider a much smaller group (or semigroup) of decohering transformations than the much larger group (semigroup) for the entire network and its global environment. This does not mean that the decoherence at one node cannot effect that of another node, which is connected via a quantum channel.

More will be said about this divide-and-conquer approach to decoherence in future papers.

II.B. Contributions to the application of quantum topology to quantum computing

II.B.1. Introduction

In this section, we summarize contributions resulting from our investigation and exploration of the application of quantum topology to quantum computing. This work includes an exploration of how a quantum computer could compute the Jones polynomial, theorems establishing that generic 4×4 solutions to the Yang-Baxter equation are universal quantum gates, relationships between topological linking and quantum entanglement, new universal gates via solutions to the Yang-Baxter equation that include the spectral parameter, new ways to understand teleportation using the categorical formalism of quantum topology, and a new theory of unitary braid group representations based on the bracket model of the Jones polynomial. These representations include the Fibonacci model of Kitaev, and promise to yield new insights into anyonic topological quantum computation.

Much of the research also involves quantum algorithms, and we have used functional integration in some of these algorithms, motivated by our on-going questions about the role of functional integration in topological quantum field theory. We are now working on deeper aspects of quantum algorithms associated with the Jones polynomial. We also expect that the abstract work that we have done on the structure of teleportation will impinge on aspects of distributed quantum computing. Finally, we expect other insights from topology in relation to the non-abelian hidden subgroup problem.

II.B.2. Quantum entanglement and topological entanglement

It is natural to ask whether there are relationships between topological entanglement and quantum entanglement. Topology studies global relationships in spaces, and how one space can be placed within another (e.g. knotting and linking of curves in three-dimensional space). Link diagrams can be used as graphical devices and holders of information. In this vein, Aravind [33] proposed that the entanglement of a link should correspond to the entanglement of a quantum state. We discussed this approach in [10, 12]. Observation at the link level is modeled by cutting one component of the link. A key example is the Borommean rings, see Figure 15.

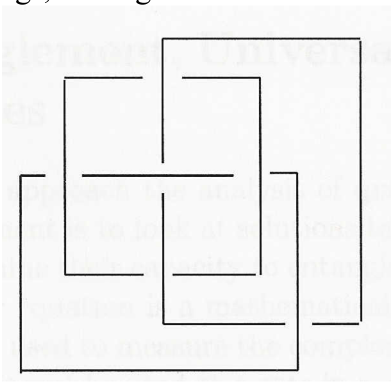


Figure 15. Borromean Rings.

Cutting any component of this link yields a remaining pair of unlinked rings. The Borommean rings are entangled (viz., the link is not split), but any two of them are unentangled. In this sense, the Borromean rings are analogous to the GHZ state $|GHZ\rangle = (|000\rangle + |111\rangle)/\sqrt{2}$. Observation of any factor (qubit) of the GHZ yields an unentangled state. Aravind points out that this property is basis dependent, and we further point out that there are states whose entanglement after an observation is probabilistic. Consider, for example, the state $(|000\rangle + |001\rangle + |101\rangle + |110\rangle)/2$. Observation in any coordinate yields an entangled or an unentangled state with equal probability. New ways to use link diagrams must be invented to map the properties of such states, see [1].

Our analysis of the Aravind analogy places it as an important question to which no definitive answer has yet been given. Our work shows that the analogy, taken literally, requires that a given quantum state would have to be correlated with a multiplicity of topological configurations. We are nevertheless convinced that the classification of quantum states according to their correspondence to topological entanglement will be of practical importance to quantum computing, distributed quantum computing and relations with quantum information protocols.

II.B.3. Entanglement, universality and unitary R-matrices

Another way to approach the analysis of quantum entanglement and topological entanglement is to look at solutions to the Yang-Baxter equation (see below) and examine their capacity to entangle quantum states. A solution to the Yang-Baxter equation is a mathematical structure that lives in two domains. It can be used to measure the complexity of braids, links and tangles, and it can (if unitary) be used as a gate in a quantum computer. We decided to investigate the quantum entangling properties of unitary solutions to the Yang-Baxter equation.

We consider unitary gates R that are both universal for quantum computation and are also solutions to the condition for topological braiding. A Yang-Baxter operator or R-matrix [**BA**] is an invertible linear operator $R : V \otimes V \to V \otimes V$, where V is a vector space, so that R satisfies the Yang-Baxter equation:

$$(R \otimes I)(I \otimes R)(R \otimes I) = (I \otimes R)(R \otimes I)(I \otimes R) \quad ,$$

where I is the identity map of V. This concept generalizes the permutation of the factors (i.e., it generalizes a swap gate when V represents one qubit).

Topological quantum link invariants are constructed by the association of an R-matrix R to each elementary crossing in a link diagram, so that an R-matrix R is regarded as representing an elementary bit of braiding given by one string crossing over another. In Figure 16 below, we have illustrated the braiding identity that corresponds to the Yang-Baxter equation. There is no room in this brief description to give the full translation from the topological picture into the algebraic one. Suffice it to say that each braiding picture with its three input lines (below) and output lines (above) corresponds to a mapping of the three fold tensor product of the vector space V to itself, as required by the algebraic equation quoted above, and the pattern of placement of the crossings in the diagram correspond to the factors $R \otimes I$ and $I \otimes R$. The point is that this crucial topological move has an algebraic expression in terms of the R-matrix R.

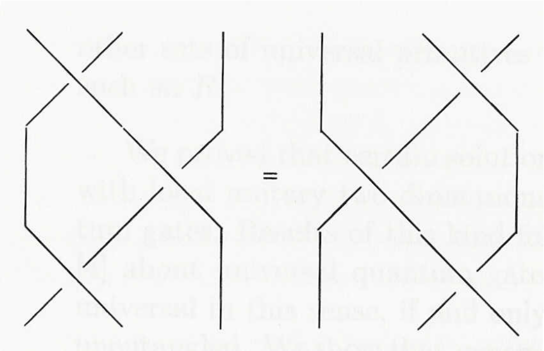


Figure 16. The Yang-Baxter Equation at the braid level.

We worked on relating topology, quantum computing, and quantum entanglement through the use of $\boldsymbol{R}$-matrices. In order to accomplish this aim, we have studied the following unitary $\boldsymbol{R}$-matrices, interpreting them as both braidings and quantum gates.

The problem of finding unitary R-matrices turns out to be surprisingly difficult. Dye (Kauffman's former graduate student) [15] has classified all such matrices of size $\mathbf{4\times4}$, and we are still working on a general theory for the classification of unitary $\boldsymbol{R}$-matrices in other dimensions.

A key question about unitary R-matrices is to understand their capability of entangling quantum states. We use the criterion that $|\varphi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ is entangled if and only if $ad-bc\neq 0$. This criterion is generalized to higher dimensional pure states in the papers [10, 11, 12, 20] by Kauffman and Lomonaco. We discovered families of $\boldsymbol{R}$-matrices that detect topological linking if and only if they can entangle quantum states. A recent example in [16] is a unitary $\boldsymbol{R}$-matrix that is highly entangling for quantum states. It takes the standard basis for the tensor product of two single-qubit spaces onto the Bell basis. On the topological side, $\boldsymbol{R}$ generates a non-trivial invariant of knots and links that is a specialization of the well-known link invariant, the Homflypt polynomial.

Entanglement and quantum computing are related in a myriad of ways, not the least of which is the fact that one can replace the $\boldsymbol{CNOT}$ gate by another gate $\boldsymbol{R}$ and maintain universality (as described above) just so long as $\boldsymbol{R}$ can entangle quantum states. That is, $\boldsymbol{R}$ can be applied to some unentangled state to produce an entangled state. It is of interest to examine other sets of universal primitives that are obtained by replacing $\boldsymbol{CNOT}$ by such an $\boldsymbol{R}$.

We proved that certain solutions $\boldsymbol{R}$ to the Yang-Baxter equation, together with local unitary two dimensional operators, form a universal set of quantum gates. Results of this kind follow from general results of the Brylinskis [39] about universal quantum gates. The Brylinskis show that a gate $\boldsymbol{R}$ is universal in this sense, if and only if it can entangle a state that is initially unentangled. We show that generically, the $\mathbf{4\times4}$ solutions to the Yang-Baxter equation are universal quantum gates.

For example, the following solutions to the Yang-Baxter equation are universal quantum gates (in the presence of local unitary transformations):

$$R=\frac{1}{\sqrt{2}}\begin{pmatrix}1&0&0&1\\0&1&-1&0\\0&1&1&0\\-1&0&0&1\end{pmatrix},\quad R'=\begin{pmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&-1\end{pmatrix},\quad \text{and}\quad R''=\begin{pmatrix}a&0&0&0\\0&0&b&0\\0&b&0&0\\0&0&0&a\end{pmatrix}$$

where a,b are unit complex numbers with $a^2\neq b^2$.

The unitary matrix R is the Bell-Basis change matrix, alluded to above. The unitary matrix R' is a close relative to the swap-gate (which is not universal). The unitary matrix R'' is both a universal gate and a useful matrix for topological purposes (it detects linking numbers). In this last example, we see a solution to the Yang-Baxter equation that detects topological linking exactly when it entangles quantum states.

These results about R-matrices are fundamental for understanding topological relationships with quantum computing, but they are only a first step in the direction of topological quantum computing. In topological quantum computing one wants to have all gates and compositions of gates interpreted as part of a single representation of the Artin Braid Group. By taking only a topological operator as a replacement for $CNOT$, we leave open the question of the topological interpretation of local unitary operators.

One must go on and examine braiding at the level of local unitary transformations and the problem of making fully topological models. The first step in this process (although made only recently by us [30]) is to classify representations of the three-strand braid group into $SU(2)$. Here one looks not for solutions to the Yang-Baxter equation, but rather for matrices A and B in $U(2)$ such that $ABA = BAB$. This is the analogue of the Yang-Baxter equation and it has many solutions in $U(2)$. Some of these pairs generate dense subsets of $U(2)$ and so can be used in principle to generate all local unitary transformations. At this stage one has a mixed topological generation of quantum computing: one type of transformation for local unitary operators and a second (Yang-Baxter) transformation for the universal 2-qubit gate.

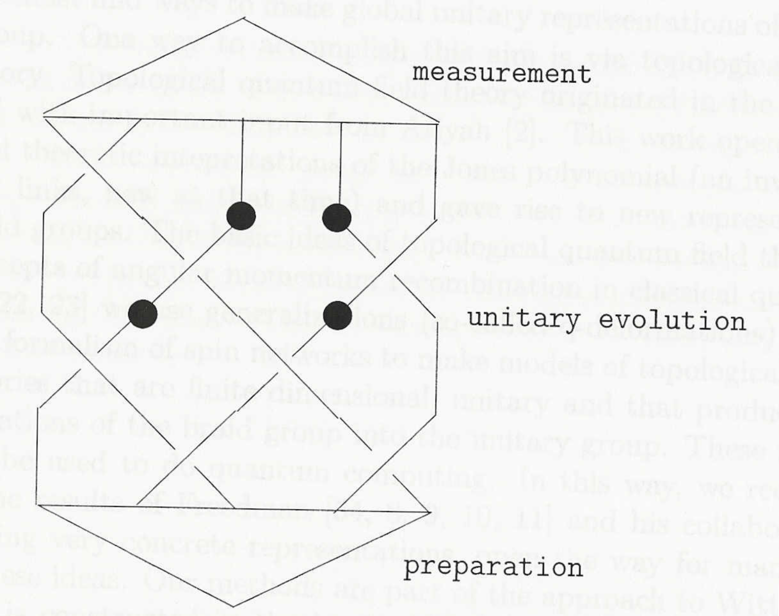


Figure 17. A topological quantum computer.

In the diagram above, we have illustrated the form of such a rudimentary topological quantum computer. The crossings represent 2-qubit gates that are solutions to the Yang-

Baxter equation. The dark circles represent local unitary transformations that may themselves be generated by a unitary representation of the braid group. To go further involves finding braiding representations into $U(2)$ that extend to dense representations in $U(N)$ for larger values of N. This is where topological quantum field theory comes into play.

In the next section we outline our approach to full topological quantum computation.

II.B.4. Topological quantum field theory and topological quantum computation

As described above, one comes to a barrier if one only attempts to construct individual topological gates for quantum computing. In order to go further, one must find ways to make global unitary representations of the Artin Braid Group. One way to accomplish this aim is via topological quantum field theory. Topological quantum field theory originated in the work of Witten [73] with important input from Atiyah [34]. This work opened up quantum field theoretic intepretations of the Jones polynomial (an invariant on knots and links, new at that time) and gave rise to new representations of the braid groups. The basic ideas of topological quantum field theory generalize concepts of angular momentum recombination in classical quantum physics. In [65, 30], we use generalizations (so-called q-deformations) of the Penrose [68] formalism of spin networks to make models of topological quantum field theories that are finite dimensional, unitary and that produce dense representations of the braid group into the unitary group. These representations can be used to do quantum computing. In this way, we recover a version of the results of Freedman [45, 46, 47, 48, 49] and his collaborators and, by making very concrete representations, open the way for many applications of these ideas. Our methods are part of the approach to Witten's invariants that is constructed in the book of Kauffman and Lins [65]. This work is directly based on the combinatorial knot theory associated with the Jones polynomial. Thus our work provides a direct and fundamental relationship between quantum computing and the Jones polynomial.

Here is a very condensed presentation of how unitary representations of the braid group are constructed via topological quantum field theoretic methods. The structure described here is sometimes called the Fibonacci model [30, 60, 69]. One has a mathematical particle with label P that can interact with itself to produce either itself labeled P or itself with the null label $*$. When $*$ interacts with P the result is always P. When $*$ interacts with $*$ the result is always $*$. One considers process spaces where a row of particles labeled P can successively interact subject to the restriction that the end result is P.

For example the space $V\left[(ab)c\right]$ denotes the space of interactions of three particles labeled P. The particles are placed in the positions a,b,c. Thus, we begin with $(PP)P$.

In a typical sequence of interactions, the first two Ps interact to produce a $*$, and the $*$ interacts with P to produce P.

$$(PP)P \to (*)P \to P.$$

In another possibility, the first two P's interact to produce a P, and the P interacts with P to produce P.

$$(PP)P \to (P)P \to P.$$

It follows from this analysis that the space of linear combinations of processes $V[(ab)c]$ is two dimensional. The two processes we have just described can be taken to be the qubit basis for this space. One obtains a representation of the three strand Artin braid group on $V[(ab)c]$ by assigning appropriate phase changes to each of the generating processes. One can think of these phases as corresponding to the interchange of the particles labeled a and b in the association $(ab)c$. The other operator for this representation corresponds to the interchange of b and c.

This interchange is accomplished by a unitary change of basis mapping

$$F : V[(ab)c] \to V[a(bc)].$$

If

$$A : V[(ab)c] \to V[a(bc)]$$

is the first braiding operator (corresponding to an interchange of the first two particles in the association), then the second operator

$$B : V[(ab)c] \to V[a(bc)]$$

is accomplished via the formula $B = F^{-1}AF$ where the A in this formula acts in the second vector space $V[(ab)c]$ to apply the phases for the interchange of b and c.

In this scheme, vector spaces corresponding to associated strings of particle interactions are interrelated by recoupling transformations that generalize the mapping F indicated above. A full representation of the Artin braid group on each space is defined in terms of the local interchange phase gates and the recoupling transformations. These gates and transformations have to satisfy a number of identities in order to produce a well-defined representation of the braid group. These identities were discovered originally in relation to topological quantum field theory. In our approach [30], the structure of phase gates and recoupling transformations arise naturally from the structure of the bracket model for the Jones polynomial. Thus we obtain a knot-theoretic basis for topological quantum computing.

Many questions arise from this approach to quantum computing. The deepest question is whether there are physical realizations for the mathematical particle interactions that constitute such models. It is possible that such realizations may come about by way of the fractional quantum Hall effect or by other means. We are working on the physical basis for such models by addressing the problem of finding a global Hamiltonian for them, in analogy to the local Hamiltonians that can be constructed for solutions to the Yang-Baxter equation. We are also investigating specific ways to create and approximate gates in these models, and we are working on the form of quantum computers based on recoupling and braiding transformations.

These models are based on the structure of the Jones polynomial [4, 61, 62, 63, 64]. They lead naturally to the question of whether or not there exists a polynomial time quantum algorithm for computing the Jones polynomial. The problem of computing the Jones polynomial is known to be classically #P-hard, and hence, classically computationally harder than NP-complete problems. Should such a polynomial time quantum algorithm exist, then it would be possible to create polynomial time quantum algorithms for any NP-complete problem, such as for example, the traveling salesman problem. This would indeed be a major breakthrough of greater magnitude than that arising from Shor's and Simon's quantum algorithms. The problem of determining the quantum computational hardness of the Jones polynomial would indeed shed some light on the very fundamental limits of quantum computation.

A polynomial time quantum algorithm (called the **Ahronov-Freedman-Jones-Kitaev-Landau (AFJKL) algorithm**) for approximating the value of the Jones polynomial $\boldsymbol{L(t)}$ at primitive roots of unity can be found in [32]. We are currently writing a paper [30] that shows that this algorithm can not successfully be extended by polynomial interpolation to a polynomial time quantum algorithm for computing the Jones polynomial. However, there is a loop hole. It may well still be possible to modify the AFJKL algorithm in such a way that it can be used to create a polynomial time algorithm for $\boldsymbol{L(t)}$. We propose to investigate why this is or is not the case. Our objective is to come to a better understanding of the exact divide between classical and quantum algorithms.

III. Epilogue

In the above section II, we have explained extensively the many contributions to quantum computing and to quantum information science resulting from this contract, as they were outlined in the executive summary given in section I. Supporting documents and supplementary material for this final report can be found at the website:

http://www.csee.umbc.edu/~lomonaco/DARPA/01-06finalrpt

IV. List of publications written under this grant

IV.A. Books published under this grant

- Lomonaco, Samuel J., Jr., (ed.), ***"Quantum Computation: A Grand Mathematical Challenge for theTwenty-First Century and the Millennium,"*** Proceedings of the Symposia of Applied Mathematics, vol. 58, American Mathematical Society, Providence, Rhode Island, (2002). (358 pages) (http://www.ams.org/bookstore?fn=20&arg1=whatsnew&item=PSAPM-58) (http://www.csee.umbc.edu/~lomonaco/ams/Lecture_Notes.html)

- Lomonaco, Samuel J., Jr., and Howard E. Brandt, (eds.), **Quantum Computation and Information,"** AMS Contemporary Mathematics, vol. 305, American Mathematical Society, Providence, RI, (2002). (310 pages) (http://www.csee.umbc.edu/~lomonaco/ams/Special.html)

IV.B. Papers published under this grant

[1] Lomonaco, Samuel L., Jr., **A Rosetta stone for quantum mechanics with an introduction to quantum computation,** AMS PSAPM/58, (2002), 3-65. (http://arxiv.org/abs/quant-ph/0007045)

[2] Kauffman, Louis H., **Biologic,** AMS CONM/304, (2002), 313 - 340. (http://arxiv.org/abs/quant-ph/0204007)

[3] Lomonaco, Samuel J., Jr., **A Talk on Quantum Cryptography, or How Alice Outwits Eve,** AMS PSAPM/58, (2002), 237-264. (http://arxiv.org/abs/quant-ph/0102016)

[4] Kauffman, Louis H., **Quantum computing and the Jones polynomial,** AMS CONM/ 305, (2002), 101-137. (http://arxiv.org/abs/math.QA/0105255)

[5] Lomonaco, Samuel J., Jr., **Shor's Quantum Factoring Algorithm,** AMS PSAPM/58, (2002), 161-179. (http://arxiv.org/abs/quant-ph/0010034)

[6] Kauffman, Louis H., **Quantum topology and quantum computing,** AMS PSAPM/58, (2002), 273-303. (http://www.math.uic.edu/~kauffman/QTandQC.pdf)

[7] Lomonaco, Samuel J., Jr., **Grover's quantum search algorithm,** AMS PSAPM/58, (2002), 181-192. (http://arxiv.org/abs/quant-ph/0010040)

[8] Kauffman, Louis H., and Samuel J. Lomonaco, Jr., **Comparing Quantum Entanglement and Topological Entanglement,** Proc. ANPA 23, (2002), 135-160. (http://arxiv.org/abs/quant-ph/0205137)

[9] Lomonaco, Samuel J., Jr., and Louis H. Kauffman, **Quantum Hidden Subgroup Algorithms: A Mathematical Perspective,** AMS CONM/ 305, (2002), 139-202. (http://arxiv.org/abs/quant-ph/0201095)

[10] Kauffman, Louis H., and Samuel J. Lomonaco, Jr., **Quantum entanglement and topological entanglement,** New Journal of Physics 4 (2002), pp. 73.1 - 73.18. (http://www.njp.org/).

[11] Lomonaco, Samuel J., Jr., **An entangled tale of quantum entanglement,** AMS PSAPM/58, (2002), 305-349. (http://arxiv.org/abs/quant-ph/0101120)

[12] Kauffman, Louis H., and Samuel J. Lomonaco, Jr., **Entanglement Criteria - Quantum and Topological,** SPIE Proceedings on Quantum Information and Computation, Vol. 5105, 11, (2003), 51-58. (http://arxiv.org/abs/quant-ph/0304091)

[13] Lomonaco, Samuel J., Jr., and Louis H. Kauffman, **Continuous quantum hidden subgroup algorithms,** SPIE Proceedings on Quantum Information and Computation, Vol. 5105, 11, (2003), 80-89. (http://arxiv.org/abs/quant-ph/0304084)

[14] Kauffman, Louis H., **Non-commutative calculus and discrete physics,** (arxiv:quant-ph/030305 v1 11 Mar 2003) in "Boundaries - Proceedings of the 24th Annual International Meeting of the Alternative Natural Philosophy Association", Wesley House, Jesus Lane, Cambridge, September 2002. Published by ANPA c/o Dr. K. Bowden, Theo. Phy. Res. Unit, Birkbeck College, Malet St., London WCIE 7HX, pp. 73-128. (http://arxiv.org/abs/quant-ph/0303058)

[15] Dye, Heather, **Unitary solutions to the Yang-Baxter equation in dimension four**, Quantum Information Processing, Vol. 2, (2003), 117-150. (http://arxiv.org/abs/quant-ph/0211050)

[16] Kauffman, Louis H., and Samuel J. Lomonaco, Jr., **Braiding Operators are Universal Quantum Gates,** New Journal of Physics, 6, (2004) 134, 1-39. (http://arxiv.org/abs/quant-ph/0401090)

[17] Kauffman, Louis H., **Non-commutative Worlds,** New Journal of Physics 6 (2004) 173, pp. 1-46. (Short version in "Spin, Proceedings of ANPA 25", Keith Bowden - editor, (2004) (http://arxiv.org/abs/quant-ph/0503198)

[18] Yimsiriwattana, Anocha , and Samuel J. Lomonaco, Jr., **Distributed quantum computing: A distributed Shor algorithm,** SPIE Proceedings on Quantum Information and Computation, (2004). (http://arxiv.org/abs/quant-ph/0403146)

[19] Lomonaco, Samuel J., Jr., and, Louis H. Kauffman, **Quantum Hidden Subgroup Algorithms: The Devil Is in the Details,** 2004 Proceedings of SPIE Proceedings on Quantum Information and Computation, (2004), 137-141. (http://arxiv.org/abs/quant-

ph/0403229)

[20] Kauffman, Louis H., and Samuel J. Lomonaco Jr., **Quantum knots,** SPIE Proceedings on Quantum Information and Computation, (2004), 268-284. (http://arxiv.org/abs/quant-ph/0403228)

[21] Kauffman, Louis H., **Biologic II,** in **"Woods Hole Mathematics"** edited by Nils tongring and R. C. Penner, World Scientific Series on Knots and Everything Vol. 34, (2004), 94-132.

[22] Kauffman, Louis H., **Non-Commutative Worlds -- A Summary,** Proceedings of ANPA, (2004). (http://arxiv.org/abs/quant-ph/0503198)

[23] Kauffman, Louis H., **Teleportation Topology,** Proceedings of the Byelorus Conference on Quantum Optics, (2004). (http://arxiv.org/abs/quant-ph/0407224)

[24] Lomonaco, Samuel J., Jr., and Louis H. Kauffman, **A Continuous Variable Shor Algorithm,** AMS CONM/381, (2005), 97-108. (http://arxiv.org/abs/quant-ph/0210141)

[25] Yimsiriwattana, Anocha, and Samuel J. Lomonaco, Jr., **Generalized GHZ States and Distributed Quantum Computing,** AMS CONM/381, (2005), 131-147. (http://arxiv.org/abs/quant-ph/0402148)

[26] Kauffman, Louis H., Yong Zhang and Mo Lin Ge, **Yang--Baxterizations, Universal Quantum Gates and Hamiltonians,** Quantum Information Processing, Vol 4. No. 3, August 2005, 159 - 197. (http://arxiv.org/abs/quant-ph/0502015)

[27] Kauffman, Louis H., Yong Zhang and Mo Lin Ge, **Universal Quantum Gates,** (to appear in IJQI - World Scientific). (http://arxiv.org/abs/quant-ph/0412095)

[28] Kauffman, Louis H., and Tomas Liko, **Knot theory and a physical state of quantum gravity,** Classical and Quantum Gravity, Vol. 23 (2006), 63-90. (http://arxiv.org/abs/hep-th/0505069)

[29] Lomonaco, Samuel J., and Louis H. Kauffman, **On quantum algorithms for the Jones polynomial,** (in preparation).

[30] Kauffman, Louis H., and Samuel J. Lomonaco Jr., **q-Deformed Spin Networks, Temperley Lieb Recoupling Theory and Anyonic Topological Quantum Computing,** (in preparation).

[31] Lomonaco, Samuel J., and Louis H. Kauffman, **Is Grover's algorithm a quantum hidden subgroup algorithm?,** (in preparation).

[32] Lomonaco, Samuel J., Jr., and Louis H. Kauffman, **Quantum hidden subgroup algorithms on free groups**, (in preparation).

V. List of PowerPoint and other presentations

Listed below are the PowerPoint presentations given at the biannual DARPA QuIST Program Review Meetings, as well as some other pertinent presentations given at professional conferences. All of these presentations can be downloaded from the website: http://www.csee.umbc.edu/~lomonaco/DARPA/01-06finalrpt

DARPA QuIST Progran Review Presentations

Talks by PI

2002-Boston-Lomonaco-QuISTPresentation.ppt

2003-Lomonaco-Poster-Ft-Lauderdale.ppt
2003-Lomonaco-QuISTBerverlyHills2003.ppt
2003-Lomonaco-QuISTBeverlyHills2003.pdf
2003-Lomonaco-UMBC-SiteRev-Mar10.ppt

2004-Lomonaco-QuIST-Arizona.ppt
2004-Lomonaco-QuISTChicago.ppt
2004-Lomonaco-FoQuS.ppt

2005-Lomonaco-QuIST-St-Augustine-FL.pdf
2005-Lomonaco-QuIST-St-Augustine-FL.ppt

Some other PowerPoint talks by PI on these topics

2003-Lomonaco-AMS-CircleAlgorithm.ppt
2003-Lomonaco-AMS-CircleAlgorithm.pdf
2003-Lomonaco-FeynmanIntegrals.ppt
2004-Lomonaco-dist-shor.ppt
2005-Lomonaco-QCrypto-URome.ppt
2005-Lomonaco-Jones-Poly-TX-AM.ppt
2005-Lomonaco-RosettaStone-UAlbany.ppt

Talks by Co-PI

2003-Kauffman-Poster-Ft-Lauderdale.pdf
2003-Kauffman-QuISTBeverlyHills.pdf
EntanglementSlides.pdf
KauffmanQuist.pdf
SlidesQCandTQFT.pdf
SlidesQuantumTopQC.zip

VI. Ph.D.'s resulting from this grant

Heather Dye -- UIC, 2003
Thesis Title: **Detection and Characterization of Virtual Knot Diagrams**

Anocha Yimsiriwattana -- UMBC, 2004
Thesis Title: **Distributed Quantum Computing**

VII. Honors received during grant

- **PI: Samuel J. Lomonaco, Jr.**

  - Visiting Senior Research Scientist -- Fall Semester, 2002
    Mathematical Science Research Institute (MSRI)
    University of California at Berkeley
    Berkeley, California

  - Senior Lagrange Fellow --- September, 2004 to August, 2005
    Institute for Scientific Interchange (ISI)
    Torino, Italy

  - Visiting Senior Research Scientist -- Fall, 2004
    Isaac Newton Institute for Mathematical Sciences
    Cambridge University
    Cambridge, England, UK

- **Co-PI: Louis H. Kauffman**

  - Visiting Researcher -- January, 2002 to August, 2002
    Stanford Linear Accelerator Laboratory (SLAC)
    Stanford University
    Palo Alto, CA

  - Visiting Professor and Visiting Researcher
    -- September, 2003 to August, 2004
    University of Waterloo, and Perimeter Institute
    Waterloo, Ontario, Canada Waterloo, Ontario, Canada

  - Visiting Researcher -- Fall, 2004
    Isaac Newton Institute for Mathematical Sciences
    Cambridge University
    Cambridge, England, UK

  - Visiting Researcher and Mathematical Tourist -- July, 2005
    Institute for Scientific Interchange (ISI)
    Torino, Italy

VIII. Acknowledgement

We wish to thank DARPA and AFRL for four years of support. It has been a pleasure serving DARPA, as well as AFRL, by participating in the QuIST Program, and by contributing to its objectives.

IX. Other References

[32] Aharonov, D., V. Jones, and Z. Landau, **On the quantum algorithm for approximating the Jones polynomial,** (2005), (http://arxiv.org/abs/quant-ph/0511096)

[33] Aravind, P.K., **Borromean entanglement of the GHZ state**, in “Potentiality, Entanglement and Passion-at-a-Distance,” R. S. Cohen et al, (eds.) Kluwer, (1997), 53-59.

[34] Atiyah, M.F., **“The Geometry and Physics of Knots,”** Cambridge University Press, (1990).

[35] Baxter, R.J., **“Exactly Solved Models in Statistical Mechanics,”** Academic Press, (1982).

[36] Becnel, Jeremy, Doctoral dissertation, (2006).

[37] Bernstein, Ethan, and Umesh Vazirani, **Quantum Complexity Theory**, SIAM Journal of Computing, Vol. 26, No. 5, (1997), 1411-1473.

[38] Biham, Eli, Ofer Biham, David Biron, Markus Grassl, and Daniel A. Lidar, **Grover's quantum search algorithm for an arbitrary initial amplitude distribution**, Phys Rev A 60, (1999), 2742-2745.

[39] Brylinski, J-L, and R. Brylinski, **Universal Quantum Gates**, in [BC] Chapman & Hall/CRC, Boca Raton, Florida, 2002, 101--116.

[40] Brylinski, R., and G. Chen, “**Mathematics of Quantum Computation,”** Chapman & Hall/CRC Press, Boca Raton, Florida, (2002).

[41] Cleve, Richard, Artur Ekert, Chiara Macchiavello, and Michele Mosca, **Quantum Algorithms Revisited**, Phil. Trans. Roy. Soc. Lond., A, (1997). (http://xxx.lanl.gov/abs/quant-ph/9708016)

[42] Eisert, J., K. Jacobs, P. Papadopoulus, and M.B. Plenio, **Optimal local implementation of non-local quantum gates,** Phys. Rev. A, 62, 052317 (2000).

[43] Ekert, Artur K.and Richard Jozsa, **Quantum computation and Shor's factoring algorithm**, Rev. Mod. Phys., 68,(1996), pp 733-753.

[44] Ettinger, Mark, and Peter Hoyer, **On Quantum Algorithms for Noncommutative Hidden Subgroups**, (1998). (http://xxx.lanl.gov/abs/quant-ph/9807029)

[45] Freedman, M., **A magnetic model with a possible Chern-Simons phase,** quant-ph/0110060v1 9 Oct 2001, (2001).

[46] Freedman, M., **Topological Views on Computational Complexity,** Documenta Mathematica - Extra Volume ICM, (1998), 453-464.

[47] Freedman, M., M. Larsen, and Z. Wang, **A modular functor which is universal for quantum computation,** quant-ph/0001108, (2000).

[48] Freedman, M. H., A. Kitaev, Z. Wang, **Simulation of topological field theories by quantum computers,** quant-ph/0001071, (2000).

[49] Freedman, M., **Quantum computation and the localization of modular functors,** quant-ph/0003128, (2000).

[50] Fulton, William, and Joe Harris, **“Representation Theory,”** Springer-Verlag, (1991).

[51] Grover, Lov K., in Proc. 28th Annual ACM Symposium on the Theory of Computation, ACM Press, new York, (1996), 212-219.

[52] Grover, Lov K., **Quantum mechanics helps in searching for a needle in a haystack**, Phys. Rev. Lett., 79(2),(1997). (http://xxx.lanl.gov/abs/quant-ph/9706033)

[53] Grover, Lov K., **A framework for fast quantum mechanical algorithms**, (http://xxx.lanl.gov/abs/quant-ph/9711043)

[54] Hall, Marshall, **“The Theory of Groups,”** Macmillan Company, (1967).

[55] Hallgren, Sean, Alexander Russell, Amnon Ta-Shma, **The Hidden subgroup problem and quantum computation using group representations**, Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, Portland, Oregon, May 2000, 627-635.

[56] Hallgren, Sean, Alexander Russell, Amnon Ta-Shma, **The Hidden subgroup problem and quantum computation using group representations**, SIAM J. Comput., Vol. 32, No. 4, (2003), 916-934.

[57] Ivanyos, Gabor, Frederic Magniez, and Miklos Santha, **Efficient quantum algorithms for some instances of the non-Abelian hidden subgroup problem**, (2001). (http://xxx.lanl.gov/abs/quant-ph/0102014)

[58] Jozsa, Richard, **Quantum factoring, discrete logarithms and the hidden subgroup problem**, IEEE Computing in Science and Engineering, (to appear). (http://xxx.lanl.gov/abs/quant-ph/0012084)

[59] Kitaev, A., **Quantum measurement and the abelian stabiliser problem**, (1995), quant-ph preprint archive 9511026.

[60] Kitaev, A., **Anyons in an exactly solved model and beyond,** arxiv:cond-mat/0506438, (2005).

[61] Jones, V.F.R., **A polynomial invariant for links via von Neumann algebras,** Bull. Amer. Math. Soc., 129, (1985), 103-112.

[62] Kauffman, L.H., **State models and the Jones polynomial**, Topology, 26, (1987), 395-407.

[63] Kauffman, L.H., **Statistical mechanics and the Jones polynomial,** AMS CONM/78, (1989), 263-297.

[64] Kauffman, L.H., **"Knots and Physics,"** World Scientific, (1991, 1994, 2001).

[65] Kauffman, L.H., **"Temperley-Lieb Recoupling Theory and Invariants of Three-Manifolds,"** Princeton University Press, Annals Studies 114, (1994).

[66] Lomonaco, Samuel J., Jr., **The non-abelian Fourier transform and quantum computation**, MSRI Streaming Video, (2000), (http://www.msri.org/publications/ln/msri/2000/qcomputing/lomonaco/1/index.html)

[67] Mosca, Michelle, and Artur Ekert, **The Hidden Subgroup Problem and Eigenvalue Estimation on a Quantum Computer**, Proceedings of the 1st NASA International Conference on Quantum Computing and Quantum Communication, Springer-Verlag, (2001). (http://xxx.lanl.gov/abs/quant-ph/9903071)

[68] Penrose, R., **Angular momentum: An approach to Combinatorial Spacetime,** "Quantum Theory and Beyond," edited by T. Bastin, Cambridge University Press, (1969).

[69] Preskill, J., **Topological computing for beginners,** (slide presentation), Lecture Notes for Chapter 9 - Physics 219 - Quantum Computation. (http://www.iqi.caltech.edu/preskill/ph219)

[70] Russell, Alexander, and Amnon Ta-Shma, **Normal Subgroup Reconstruction and Quantum Computation Using Group Representations**, STOC, (2000).

[71] Shor, Peter W., **Polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer**, SIAM J. on Computing, 26(5) (1997), pp 1484 - 1509. (http://xxx.lanl.gov/abs/quant-ph/9508027)

[72] Shor, Peter W., **Introduction to quantum algorithms**, AMS PSAPM/58, (2002), 143-159. (http://xxx.lanl.gov/abs/quant-ph/0005003)

[73] Witten, E., **Quantum field theory and the Jones polynomial**, Commun. Math. Phys., 121, (1989), 351-399.

[74] Vazirani, Umesh, **On the power of quantum computation**, Philosophical Tranactions of the Royal Society of London, Series A, 354:1759-1768, August 1998.

[75] Vazirani, Umesh, **A survey of quantum complexity theory**, AMS PSAPM/58, (2002), 193-217.

[76] van Dam, Wim, and Lawrence, Ip, **Quantum Algorithms for Hidden Coset Problems**, (http://www.cs.caltech.edu/~hallgren/hcp.pdf)

[77] Zalka, Christof, **Grover's quantum searching algorithm is optimal**, Phys. Rev. A, Vol. 60, No. 4, (1999), 2746-2751. (http://xxx.lanl.gov/abs/quant-ph/9711070)